

abc

N° 30

COURS
D'INFORMATIQUE
PRATIQUE
ET FAMILIALE

INFORMATIQUE



L'ordinateur nu

Logiciels de gestion

Les branchés

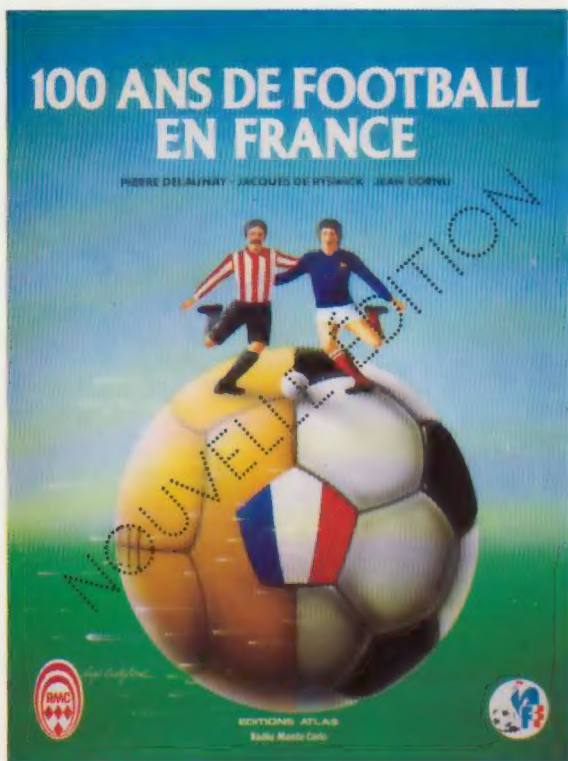
Le SED Dragon

EDITIONS
ATLAS

M6062-30-12F

85FB-3,80FS- $\$$ 1.95

Dans toutes les librairies



100 ans de football en France

Des premiers échanges de balles au bois de Boulogne jusqu'aux récentes compétitions précédant le Championnat d'Europe organisé par la France, cet ouvrage retrace l'épopée centenaire du football en France.

Une iconographie riche, le plus souvent inédite, un texte signé par trois des meilleurs spécialistes de la question font de *100 ans de football en France* un livre documenté des plus attrayants, qui satisfera tous les passionnés du ballon rond.

Un volume relié, sous jaquette illustrée.

336 pages.

478 photos en noir et blanc et en couleurs.

Format : 22,5 x 29 cm.

Dans toutes les librairies

RÉUSSIR SES PHOTOS



Réussir ses photos

Une scène vous séduit, vous la prenez en photographie! Mais trop souvent le résultat est décevant. Grâce à *Réussir ses photos*, vous saurez comment régler tous les problèmes techniques, de la mise au point à l'exposition. Vous serez en mesure de réaliser des compositions originales et de choisir judicieusement votre angle de prise de vue. *Réussir ses photos* vous apprendra à découvrir dans la nature une source inépuisable d'images exceptionnelles.

Un volume relié. Couverture illustrée.

96 pages.

203 photos et 27 schémas en couleurs et en noir et blanc.

Format : 21 x 27,5 cm.



Édité par ÉDITIONS ATLAS s.a., tour Maine-Montparnasse, 33, avenue du Maine, 75755 Paris Cedex 15. Tél. : (37) 35-40-23. Services administratifs et commerciaux : 3, rue de la Taye, 28110 Lucé. Tél. : (37) 35-40-23.

Belgique : ÉDITIONS ATLEN s.a., Bruxelles.

Canada : ÉDITIONS ATLAS CANADA Ltée, Montréal Nord.

Suisse : FINABUCH s.a., ÉDITIONS TRANSALPINES, Mezzovico.

Réalisé par EDENA s.a., 29, boulevard Edgar-Quinet, 75014 Paris.

Direction éditoriale : J.-Fr. Gautier. Service technique et artistique : F. Givone et J.-Cl. Bernar. Iconographie : J. Pierre. Correction : B. Noël.

Publicité : Anne Cayla. Tél. : 202-09-80.

VENTE AU NUMÉRO

Les numéros parus peuvent être obtenus chez les marchands de journaux ou, à défaut, chez les éditeurs, au prix en vigueur au moment de la commande. Ils resteront en principe disponibles pendant six mois après la parution du dernier fascicule de la série. (Pour toute commande par lettre, joindre à votre courrier le règlement, majoré de 10 % de frais de port.)

Pour la France, s'adresser aux services commerciaux des ÉDITIONS ATLAS. Tél. : (37) 35-40-23.

Pour les autres pays, s'adresser aux éditeurs indiqués ci-dessous.

SOUSCRIPTION

Les lecteurs désirant souscrire à l'ensemble de cet ouvrage peuvent s'adresser à :

France : DIFFUSION ATLAS, 3, rue de la Taye, 28110 Lucé. Tél. : (37) 35-40-23.

Belgique : ÉDITIONS ATLEN s.a., 55, avenue Huart-Hamoir, 1030 Bruxelles. Tél. : (02) 242-39-00. Banque Bruxelles-Lambert, compte n° 310-0018465-24 Bruxelles.

Canada : ÉDITIONS ATLAS CANADA Ltée, 11450 boulevard Albert-Hudon, Montréal Nord, H 1G 3J9.

Suisse : FINABUCH s.a., ÉDITIONS TRANSALPINES, zona industriale 6849 Mezzovico-Lugano. Tél. : (091) 95-27-44.

A NOS LECTEURS

ABC Informatique ne paraîtra pas vendredi prochain ni le vendredi suivant. Nous vous donnons donc rendez-vous le 17 août prochain.

Les éditeurs

En vente tous les vendredis. Volume III, n° 30.

ABC INFORMATIQUE est réalisé avec la collaboration de Trystan Mordrel (secrétariat de rédaction), Jean-Pierre Bourcier (coordination), Patrick Bazin, Jean-Paul Mourlon, Claire Rémy (traduction), Ghislaine Goullier (fabrication), Marie-Claire Jacquet (iconographie), Patrick Boman (correction).
Crédit photographique, couverture : Matra.

Directeur de la publication : Paul Bernabeu. Imprimé en Italie par I.G.D.A., Officine Grafiche, Novara. Distribution en France : N.M.P.P. Tax. Dépôt légal : juillet 1984. 27847. Dépôt légal en Belgique : D/84/2783/27.

© Orbis Publishing Ltd., London.
© Editions Atlas, Paris, 1984.

A NOS LECTEURS

En achetant chaque semaine votre fascicule chez le même marchand de journaux, vous serez certain d'être immédiatement servi, en nous facilitant la précision de la distribution. Nous vous en remercions d'avance.

Les Éditions Atlas



Les branchés

Le monde de la communication ne se contente plus du téléphone, trop souvent occupé. Le dialogue par micro-ordinateur et écran interposés se développe sur des réseaux toujours plus nombreux.



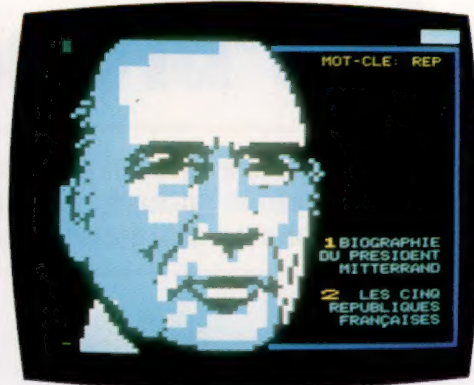
Même si le micro-ordinateur draine à lui une frange du public uniquement préoccupée de se conformer à la mode, il semble certain maintenant qu'il est promis à un avenir brillant. En effet, non seulement le matériel informatique, notamment les micro-ordinateurs, ne cesse de se perfectionner, mais, surtout, les perspectives, tantôt souhaitées tantôt redoutées, qu'offre une informatisation croissante de la société, nous disent combien ce phénomène est irréversible et nous permettent déjà d'entrevoir les bouleversements en profondeur qui se produiront dans notre vie quotidienne. L'arrivée des terminaux Minitel dans les foyers est d'ailleurs commencée.

A l'origine de cette grande mutation, la naissance et le développement fulgurant de deux inventions : le téléphone et l'informatique. L'enfant, naturel et attendu, de ce mariage est la télématique, qui trouve rapidement avec le micro-ordinateur un moyen privilégié de se diffuser dans des groupes de plus en plus divers et élargis de la société.

La grande découverte de la télécommunication a été de transformer des vibrations mécaniques en vibrations électriques ayant le même comportement, et réciproquement. Du téléphone « à ficelle » que l'on attribue au physicien anglais Hooke au téléphone électrique, et

Soyez « Télétel »

Il est possible de raccorder un micro-ordinateur sur le service Télétel. Pour cela, il faut qu'il soit compatible. S'il ne l'est déjà, il vous faut acheter un modem ou un coupleur acoustique et le logiciel correspondant au système. (Cl. SIC-PTT.)



Comme dans tous les services télématiques qui se mettent en place, vous faites appel, en premier, à un menu qui vous permet de choisir la « page » que vous souhaitez regarder. (Cl. Goupil/SIC-PTT/SAU.)

jusqu'à l'essor prodigieux des télécommunications aujourd'hui, les questions centrales restent les mêmes : comment triompher de la distance (l'Américain Bell réalise la première transmission à distance — 9 km entre Boston et Malden — en 1875 en utilisant les fils du télégraphe électrique), et rendre de plus en plus performant et maniable le matériel utilisé afin de transmettre en un minimum de temps le maximum d'informations ? Comment donc rendre le plus rapide, le plus économique et le plus efficace possible le réseau téléphonique ?

Le réseau commuté

Le premier réseau téléphonique est créé à New York en 1877, tandis que la reine Victoria dote son palais de Buckingham de cette merveille toute nouvelle; et c'est l'Exposition universelle de 1878 à Paris qui voit vraiment l'explosion du téléphone.

En 1881 sont déjà installés à Paris sept bureaux centraux (plus de trois cents abonnés), et, dans sept villes de province, des réseaux sont constitués. La première liaison « ville à ville » est établie en France entre Rouen et Le Havre en 1885, la première liaison internationale, entre Paris et Bruxelles, en 1887.

Mais cette liaison réalisée poste à poste était insuffisante; chaque abonné était alors contraint, dans un réseau de dix abonnés, de posséder neuf postes... Il fallait très vite améliorer la commutation, c'est-à-dire l'ensemble des opérations qui permettent le raccordement d'une ligne d'abonné à une autre ligne d'abonné, afin de développer et de faciliter la mise en relation téléphonique. A la commutation manuelle (dans laquelle l'opératrice doit procéder à une succession d'opérations : exploration, présélection, enregistrement, traduction, etc.) succédera la commutation automatique électromécanique dans laquelle les opérations évoquées ci-dessus seront cette fois assurées en automatique et non plus manuellement. Un plan de numérotation nationale fut adopté en 1953 pour faciliter cette transformation du réseau.

C'est maintenant la commutation électronique qui est en plein essor. Elle s'est vraiment installée en France en 1978 avec la mise en service du réseau Platon et l'expérimentation du

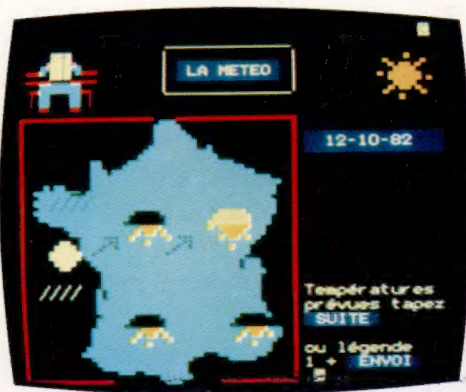
central Périclès à Clamart. De la ligne téléphonique à deux fils au câble, puis au câble coaxial, jusqu'aux satellites et au laser, un progrès considérable a été accompli, progrès qui a permis de faire fi de la distance et d'acheminer simultanément des milliers de communications téléphoniques.

Combinée à l'informatique, la télécommunication ouvre donc un vaste champ d'explorations et d'applications.

Le traitement automatique de l'information a fait lui aussi ces dernières années un fulgurant bond en avant. La sophistication de plus en plus poussée du matériel, les progrès de la miniaturisation, les recherches faites en matière de conception de logiciels et la politique adoptée par les États qui ont pris conscience de l'importance de l'« enjeu informatique » font maintenant de l'ordinateur un « meuble familial ». Sur quatre cent mille micro-ordinateurs actuellement installés en France, environ la moitié serait utilisée à des fins domestiques.

Le développement de la transmission à distance d'informations reçues ou émises par un système informatique intéresse un nombre croissant d'utilisateurs : entreprises, collectivités diverses et, bien sûr, simples particuliers. Grâce à la mise en place de réseaux de transmissions de données, à l'extension et à la modernisation du réseau téléphonique et aux progrès réalisés en matière d'informatique et d'électronique (arrivée sur le marché d'appareils à des prix plus compétitifs, efforts pour la création de logiciels, notamment des logiciels français, miniaturisation des composants...), on a vu pénétrer dans les familles depuis quelques années des appareils de petit format aux multiples usages et d'utilisation relativement simple dénommés Minitel.

Voulez-vous essayer le terminal Minitel ? Vous louerez aux P.T.T., à moins que vous n'habitiez une région déjà desservie par l'annuaire électronique — en ce cas vous en disposerez à titre gracieux —, un petit terminal beige très maniable (il pèse moins de 5 kg) doté d'un clavier repliable et d'un écran noir et blanc de 23 cm. Il est directement connectable au réseau téléphonique normal et vous permet des dialogues en langage clair. De ce fait, aucune connaissance particulière en informatique n'est demandée pour son utilisation, et le dialogue et



la procédure d'accueil sont suffisamment explicites pour guider l'utilisateur et l'aider à affiner ses questions (ce qui augmente la rapidité et l'efficacité de la recherche).

Il vous permet d'utiliser un nouveau moyen de communication, Télétel, outil multiservices, qui donne accès à un nombre illimité de centres serveurs (ordinateurs dispensant des services).

Les réseaux

Pour se « brancher » sur le centre serveur, l'utilisateur du terminal peut donc utiliser simplement le réseau téléphonique commuté; en ce cas la tarification sera fonction de la distance et du temps.

Mais il peut aussi passer par le réseau national Transpac, réseau spécifique de transmissions de données, conçu pour une utilisation optimale des artères de transmission (une communication n'utilise les ressources de transmission que lorsque des données sont transmises : il y a donc optimisation des moyens de transmission et économie importante pour l'utilisateur). Un réseau comme Transpac répond à un besoin précis : constituer un réseau de lignes téléphoniques exclusivement destinées à relier entre eux des ordinateurs spécialisés. Dans ce cas la tarification, indépendante de la distance, est celle de la taxe de base d'une communication locale. Mais les centres serveurs proposent bien sûr des services payants. Regroupant en général un grand nombre de demandeurs, ils proposent des abonnements à l'heure dont le tarif est fonction de la tranche horaire d'utilisation (heures de pointe ou heures creuses).

D'autres réseaux de transmission de données sont actuellement en fonction, comme le réseau Caducée ou le réseau Euronet, résultat d'une coopération au niveau de la Communauté européenne, qui permet aux utilisateurs européens d'accéder aux bases et banques de données scientifiques, techniques et socio-économiques des pays de la Communauté.

Les terminaux Télétel (baptisés Minitel) proposés par l'Administration offrent donc de multiples possibilités mais restent des appareils « non intelligents », incapables de mémoriser ou de raisonner. D'autres constructeurs commercialisent aussi des terminaux aux performances proches. Simples d'utilisation, aptes à rendre de

nombreux services, ils ne peuvent cependant pas remplacer les micro-ordinateurs.

De nombreux constructeurs proposent maintenant des micro-ordinateurs de faible encombrement et d'un prix relativement « raisonnable » (entre 1 500 et 6 000 F).

Composés d'un clavier et d'un écran, proposant un choix de logiciels qui leur permet d'assumer des tâches diverses, ils procurent à un utilisateur fraîchement initié à l'informatique le plaisir de « pianoter » d'abord, puis d'acquérir une maîtrise plus grande de l'outil informatique, jusqu'à ce qu'il arrive un jour à écrire lui-même les logiciels dont il a besoin. Aussi, les logiciels de calcul, de traitement de texte, les logiciels graphiques, les logiciels de gestion de fichiers... ou plus simplement les logiciels de jeux offrent des possibilités très diversifiées au propriétaire d'un micro-ordinateur qui ne désire ni approfondir ses connaissances informatiques ni confectionner lui-même ses propres logiciels.

Mais un usage plus vaste du micro-ordinateur est proposé à ceux qui relieront celui-ci au réseau téléphonique grâce à un modem ou à un coupleur acoustique et pourront de cette manière communiquer avec un centre serveur ou un autre ordinateur.

Le modem (ce mot est issu de la contraction de « modulateur-démodulateur »), comme le coupleur acoustique, appareil moins sophistiqué, permet d'harmoniser la communication entre les deux protagonistes du dialogue. En effet, la voix humaine et l'ordinateur émettent des messages à des cadences très différentes, ultrarapide pour l'ordinateur, beaucoup plus lente pour la voix humaine. Pour qu'il y ait bonne réception du message à chaque bout de la chaîne, la présence d'un intermédiaire, traducteur-transformateur en quelque sorte, est nécessaire.

En dialoguant avec un autre ordinateur, on peut alors échanger des programmes, des fichiers, des tableaux. En connectant son micro-ordinateur à un centre serveur grâce au réseau Transpac, on accède à une quantité de services spécialisés, en particulier à de nombreuses banques de données, juridiques, économiques, de sciences exactes ou de sciences humaines; grâce au lien que crée le modem, l'accès rapide à l'ensemble des connaissances stockées et gérées par les gros ordinateurs devient possible.

Le système Télétel est réalisé de telle façon qu'il vous guide systématiquement dans vos recherches, qui peuvent aller des informations météo aux jeux vidéo. (Cl. TDF/DGT/SIVEA.)



Chariot de feu

Le système d'exploitation de disquettes (SED) Dragon présente des commandes faciles d'utilisation en un BASIC de type Advanced Colon de Microsoft. Il est possible de connecter jusqu'à quatre lecteurs.

Les lecteurs Dragon utilisent des disquettes 5 pouces, simple densité. Celles-ci sont formatées magnétiquement par une commande, sous forme de 40 pistes, chacune répartie en 18 secteurs, et chaque secteur pouvant stocker 256 octets. Ce qui donne comme capacité totale de stockage sur disque : $40 \times 18 \times 256 = 184\ 320$ octets ou 180 K. Le système d'exploitation prend une partie de cet espace mémoire pour l'identification des fichiers, la gestion du disque et le répertoire, laissant à peu près 175 K disponibles. Le répertoire est la liste des fichiers du disque. En réponse à une commande, il affiche leur nom, leur type, leur nombre d'octets et le nombre d'octets encore inoccupés. Il existe quatre types de fichiers : ceux de programme BASIC (BAS) de données (DATA), les fichiers binaires (BINary) et ceux de recopie (BAckup). Chaque fichier est indiqué par une abréviation : .BAS, .DAT, .BIN, .BAK. Le système d'exploitation du Dragon n'indique pas l'emplacement en mémoire des fichiers, ni comment y accéder. Mais, du fait qu'il n'y a pas de commande pour réserver des espaces mémoire disponibles, il

existe probablement une forme de table d'allocation des blocs.

Curieusement, le système d'exploitation du Dragon ne permet pas l'accès direct aux données, un octet à la fois. C'est pourtant très utile notamment pour un programme qui a besoin d'aller chercher fréquemment de petites quantités de données — une base de données par exemple. En revanche, il existe un mode d'accès dit « accès direct simulé », qui fait à peu près la même chose avec quelques lignes de programme BASIC supplémentaires pour l'implémenter. L'implémentation du système d'exploitation du Dragon est remarquablement complémentaire avec le BASIC.

Un prix d'achat relativement élevé pour une capacité de stockage assez moyenne fait de ce lecteur de disques un lecteur d'un intérêt moyen. C'est peut-être pour cela que Dragon envisage maintenant d'élargir sa gamme de lecteurs avec un lecteur 80 pistes à double face et double densité à un prix compétitif. Un système d'exploitation résident en RAM et chargé depuis un disque est prévu pour le 64.



Langages en options

Le premier lot de logiciels du Dragon fonctionne sous le système d'exploitation OS9. Il comprend des langages avancés tels que le PASCAL et le compilateur C.

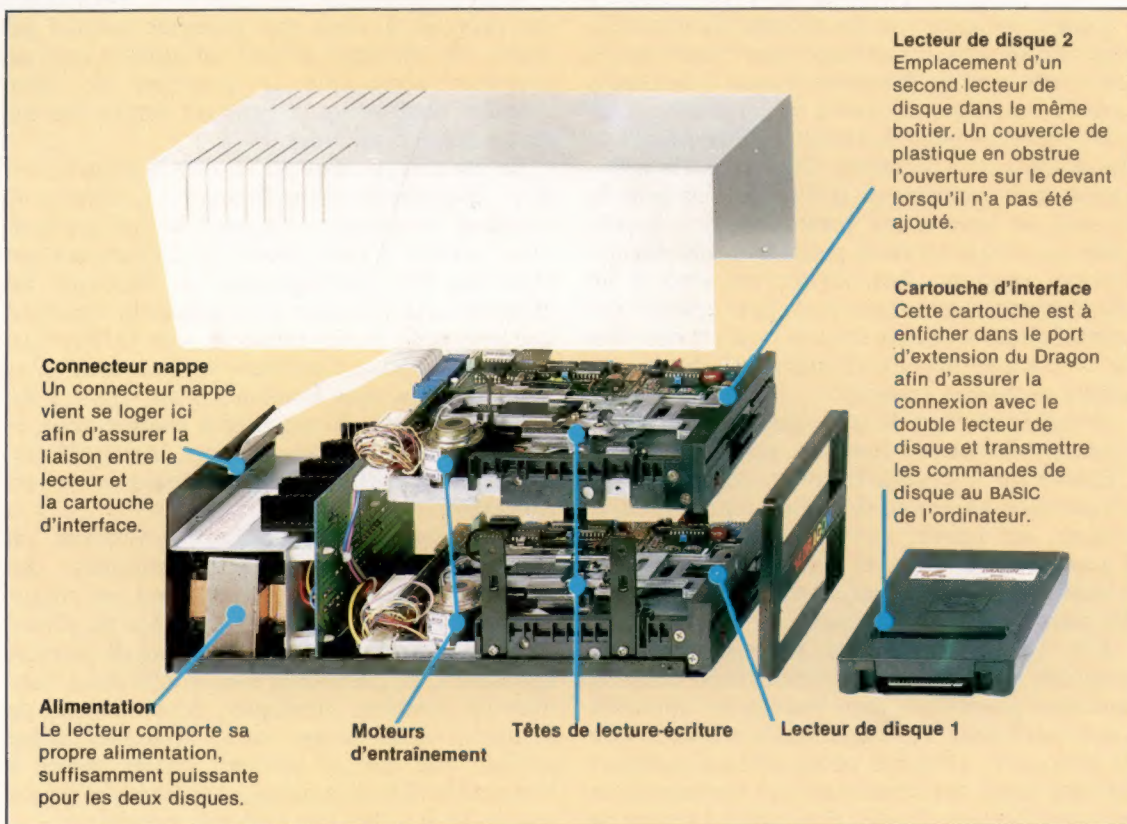


Progiciel sophistiqué

OS9 vous permet également d'exploiter des progiciels de gestion très performants. Le progiciel de traitement de texte montré ici comporte un lexique de 42 000 mots pour vérification de l'orthographe.

Fonctionnement

Les lecteurs Dragon sont livrés sous un seul boîtier pour deux lecteurs. Ils comportent leur propre alimentation. Un câble de type « nappe » les relie à une cartouche d'interface que l'on enfiche dans le Dragon. Les disques fonctionnent sous leur propre système d'exploitation, ou encore par l'intermédiaire de logiciels système d'exploitation de disque, OS9.



Lecteur de disque 2
Emplacement d'un second lecteur de disque dans le même boîtier. Un couvercle de plastique en obstrue l'ouverture sur le devant lorsqu'il n'a pas été ajouté.

Cartouche d'interface
Cette cartouche est à enficher dans le port d'extension du Dragon afin d'assurer la connexion avec le double lecteur de disque et transmettre les commandes de disque au BASIC de l'ordinateur.

Connecteur nappe
Un connecteur nappe vient se loger ici afin d'assurer la liaison entre le lecteur et la cartouche d'interface.

Alimentation
Le lecteur comporte sa propre alimentation, suffisamment puissante pour les deux disques.

Moteurs d'entraînement

Têtes de lecture-écriture

Lecteur de disque 1



Commandes de disque du Dragon

Les paramètres des commandes sont souvent identiques, le format le plus courant étant :

COMMAND«D:NOM .TYP»

D sélectionne un lecteur (de 1 à 4) et vient remplacer le lecteur par défaut. **NOM** est mis pour **NOM DU FICHIER** et peut comporter jusqu'à huit caractères. **.TYP** identifie le type du fichier. Il est optionnel, et sera en cas d'absence : **.BAS**. Nous ferons allusion à ce format par l'abréviation **FSP** (File Specification).

DRIVE N (lecteur)

N peut prendre ici une valeur comprise entre 1 et 14. Cette commande assigne le lecteur par défaut.

DSKINIT (DiSK INItialization)

Initialise le disque spécifié selon la syntaxe :

DSKINIT D,S,T

D sélectionne ici le lecteur (Drive); **S**, la face du disque (Side 1 ou 2); **T**, le nombre de pistes à formater (40 ou 80; 40 par défaut). Pour un lecteur simple (un seul disque), il faut taper **DSKINIT**(retour chariot), le disque présent sera effacé s'il y a lieu, et formaté.

DIR D (DiREctory, « répertoire »)

Affiche le répertoire du lecteur spécifié **D**, de la sorte :

```
DIR
NOMPROG.BAS 1654
M/C BINAIRE.BIN 1389
DONNEES.DAT 2581
NOMPROG.BAK 1654
167322 OCTETS LIBRES
```

Fichiers programmes

SAVE

Sauvegarde sur disque des programmes ou des fichiers binaires. **SAVE FSP** sauvegarde un fichier programme. Il n'est pas nécessaire de spécifier l'extension **.BAS** puisque c'est la valeur par défaut. La commande suivante sauvegarde un fichier binaire :

SAVE FSP,SSSS,EEEE,XXXX

SSSS (Start) est ici l'adresse en décimal du début du code à sauvegarder; **EEEE** (End) est l'adresse de fin de ce code; et **XXXX** est l'adresse à partir de laquelle le programme est exécuté.

LOAD(chargement)

Vient chercher sur le disque des fichiers programmes ou binaires. La commande est de la forme :

LOAD FSP

Lorsque le fichier spécifié est de type binaire, on peut remplacer **FSP** par **SSSS**. Cela indiquera la nouvelle adresse mémoire de début de fichier.

RUN FSP (exécution)

Charge et exécute immédiatement le programme **BASIC** spécifié.

CHAIN FSP (chaînage)

Charge et exécute un programme sans modifier les variables préalablement stockées. Cette commande est très utile pour les programmes qui se partagent les mêmes données. Si vous ajoutez **SSSS**, le résultat sera le même que précédemment.

FREE D

Affiche le nombre d'octets libres sur le lecteur spécifié.

COPY

Copie le fichier **ANCFB** (ANCIen), sous la forme **NOVFSB** (NOUveau) :

COPY ANCFB TO NOVFSB

Si les numéros de lecteur ne sont pas indiqués, la copie a lieu sur le même disque mais sur le lecteur par défaut.

RENAME

Change le nom d'un fichier sans modifier son type. L'ancienne et la nouvelle spécification (**FSP**) doivent s'appliquer au même lecteur, ou être toutes les deux assignées par défaut.

MERGE FSP

Fusionne deux fichiers de programme **BASIC** dont l'un se trouve en mémoire. S'il y a conflit entre des numéros de lignes des deux programmes, le programme précédemment sur disque aura la priorité.

KILL FSP

Détruit sur le disque le fichier spécifié.

PROTECT

Cette commande protège un programme contre tout risque d'effacement ou d'« écrasement en écriture ». Seule la commande d'initialisation échappe à cette protection. La syntaxe est la suivante :

PROTECT ON FSP

Cela suscite également l'affichage d'une zone mémoire annexée au fichier protégé, **P** (Protect), lors d'une demande de répertoire. Pour retirer la protection, on utilisera la commande : **PROTECT OFF FSP**.

BACKUP

Copie l'intégralité d'un disque d'un lecteur (**DA**, Drive A) sur un autre (**DB** - Drive B) :

BACKUP DA TO DB,S,T

S et **T** sont les mêmes que pour la commande **DISKINIT**. Ils permettent de réaliser des copies selon un format différent et sur un disque différent de celui de départ.

VERIFY

ON active et **OFF** inhibe cette commande qui vérifie automatiquement que le contenu du fichier sauvegardé est bien le même que celui d'origine.

Fichiers de données

FWRITE (File WRITE, « fichier en écriture ») Sert à créer un fichier de données et à écrire dans des listes de variables. Pour chaque commande **FWRITE**, un canal est ouvert vers le fichier spécifié, permettant d'ajouter ultérieurement d'autres données. La syntaxe est la suivante :

FWRITE«NOM»;VAR

NOM est mis pour **NOM DU FICHIER** et **VAR** représente la liste de variables qui comportent les données à sauvegarder.

FWRITE«NOM»,FROM SB,FOR
TB;VAR

Cette commande écrit **VAR** sur le fichier identifié par **NOM**, à partir de l'octet **SB**, sur une longueur de **TB** octets. On ne peut ouvrir simultanément que dix fichiers par **FWRITE**.

CLOSE D

Ferme les canaux de fichiers qu'avaient ouverts **FWRITE**, **FREAD** et **FLREAD** vers un lecteur déterminé.

CREATE«NOM»,FL

Crée un fichier de données, identifié par **NOM**, de **FL** octets de long.

FREAD

Commande construite comme **FWRITE**. **VAR** est trouvé en mémoire, ou, pour le deuxième exemple donné, **VAR** est lu à partir de l'octet **SB**. Le pointeur de lecture est alors avancé de **SB+TB** octets.

FLREAD

Même syntaxe que pour **FREAD**, si ce n'est que les virgules et les deux-points ne sont pas considérés comme terminateurs.

EOF(End Of File, « fin de fichier »)

Indique le dernier enregistrement valide du fichier en cours de lecture. Par exemple :

EP=EOF«NOM»

EP est égal à zéro jusqu'à ce que le pointeur de lecture rencontre le dernier enregistrement et donne à **EP** la valeur 1.

LOC«NOM»

Affiche la position du pointeur de lecture en tant que numéro du prochain octet à être lu pour le fichier spécifié.

SWRITE

Cette commande permet de stocker des données dans le secteur **S**, sur la piste **T**, comme les deux chaînes **A\$** et **B\$**, à concurrence d'un maximum de 128 octets chacune. La syntaxe est la suivante :

SWRITE D,T,S,A\$,B\$

SREAD

Restitue les données stockées par **SWRITE** selon la même syntaxe. **A\$** et **B\$** peuvent recevoir des noms nouveaux de variables.

Chemins logiques

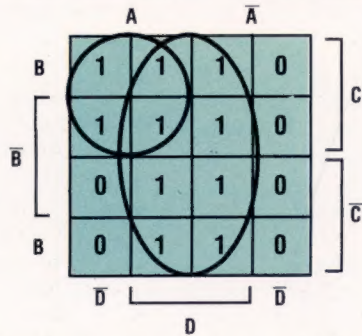
Nous avons vu précédemment l'utilisation du tableau de Karnaugh pour simplifier des expressions d'algèbre de Boole. Nous abordons maintenant des cas plus complexes à quatre variables.

Quatre variables : quand il s'agit d'expressions booléennes comprenant quatre variables, le tableau de Karnaugh et les expressions elles-mêmes, cela peut sembler très difficile. Mais, en appliquant scrupuleusement les principes élémentaires dégagés des tableaux de Karnaugh à deux ou à trois variables, les choses s'arrangent d'elles-mêmes.

Prenons l'expression suivante à simplifier :

$$ABC\bar{D} + ABCD + \bar{A}BCD + A\bar{B}CD + \bar{B}CD + A\bar{B}CD + \bar{A}BCD + BCD$$

Nous voyons qu'un tableau de Karnaugh à quatre variables est nécessaire. Bien qu'il y ait huit éléments dans cette expression, nous en avons en réalité dix (BCD et $\bar{B}CD$ représentent chacun deux éléments). Voici le tableau :



Nous constatons à partir du tableau que le groupe central de huit 1 représentent toutes les combinaisons possibles faisant intervenir D. Le groupe de quatre 1 en haut à gauche réunit tous les cas possibles pour A ET C. Aussi l'expression peut-elle se simplifier en A ET C OU D (ce qui s'écrit : $A.C + D$).

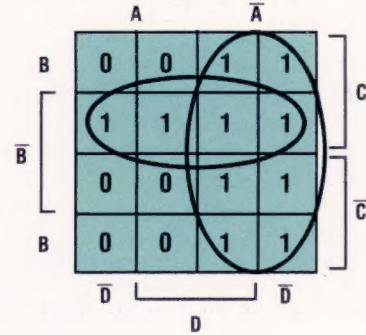
Une expression peut quelquefois avoir besoin d'être modifiée pour pouvoir être représentée sous forme d'un tableau de Karnaugh, comme dans l'exemple suivant :

$$A + B + C + \bar{A}.B + \bar{B} + \bar{C}$$

Appliquons d'abord la loi de Morgan à l'expression avant de voir s'il est possible de dresser le tableau de Karnaugh :

$$\bar{A}.\bar{B}.\bar{C} + \bar{A}.B + \bar{B}.C$$

Ce qui donne :



A partir du tableau de Karnaugh, l'expression se simplifie ainsi : $\bar{A} + \bar{B}.C$. En appliquant à nouveau la loi de Morgan, on obtient :

$$A.(B + C)$$

Création de circuit logique

Exemple 1 : 30 jours

Codons chaque mois de l'année par un nombre binaire à quatre bits : de 0001 pour janvier jusqu'à 1100 pour décembre. Nous voulons créer un circuit qui accepte en entrée le code à quatre bits, et qui produise en sortie le code 1 lorsque le mois concerné a trente jours.

La table de vérité sera :

MOIS	ENTRÉES				SORTIES
	A	B	C	D	
	0	0	0	0	X
Janv.	0	0	0	1	0
Févr.	0	0	1	0	0
Mars	0	0	1	1	0
Avr.	0	1	0	0	1
Mai	0	1	0	1	0
Juin	0	1	1	0	1
Juil.	0	1	1	1	0
Août	1	0	0	0	0
Sept.	1	0	0	1	1
Oct.	1	0	1	0	0
Nov.	1	0	1	1	1
Déc.	1	1	0	0	0
	1	1	0	1	X
	1	1	1	0	X
	1	1	1	1	X

Le résultat X en sortie signifie une mauvaise entrée de données. Nous supposons que le circuit ne recevra pas de tels signaux. Nous pouvons établir l'expression booléenne suivante pour les données entrées, à partir des cas où $S = 1$:

$$S = \bar{A}.B.\bar{C}.\bar{D} + \bar{A}.B.C.\bar{D} + A.\bar{B}.\bar{C}.D + A.B.C.D$$

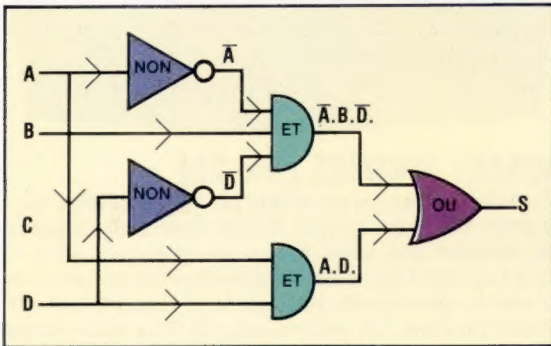
Nous pouvons alors remplir le tableau de Karnaugh en incluant les conditions d'entrée erronées (X) :

	A	\bar{A}	
B	X	X	0
\bar{B}	0	1	0
B	0	1	0
\bar{B}	0	X	0
B	0	0	1
	\bar{D}	D	\bar{D}

Et ce tableau nous permet de simplifier l'expression pour aboutir à :

$$A.D + \bar{A}.B.\bar{D}$$

Le circuit logique dit « de trente jours » sera donc :



Exemple 2 : nombres impairs

Sachant que les nombres compris entre 0 et 15 peuvent être codés en binaire sur quatre chiffres (de 0000 à 1111), nous devons établir un circuit logique qui prendrait en entrée le code en question et produirait 1 en sortie lorsque le résultat serait un nombre impair supérieur à deux. Écrivons la table de vérité recensant les possibilités :

NOMBRE DÉCIMAL	ENTRÉES				SORTIES S
	A	B	C	D	
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	1

A partir de cette table de vérité, nous pouvons écrire l'expression booléenne suivante, pour toutes les occurrences vérifiant S (S vrai) :

$$S = \bar{A}.\bar{B}.C.D + \bar{A}.B.\bar{C}.D + \bar{A}.B.C.D + A.\bar{B}.\bar{C}.D + A.\bar{B}.C.D + A.B.\bar{C}.D + A.B.C.D$$

Le tableau de Karnaugh donnera :

	A	\bar{A}	
B	0	1	1
\bar{B}	0	1	1
B	0	1	0
\bar{B}	0	1	0
B	0	1	0
	\bar{D}	D	\bar{D}

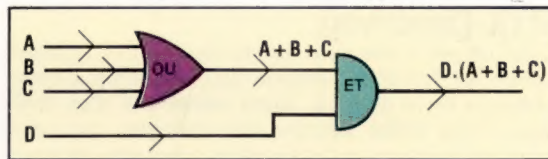
A partir de ce tableau, nous pouvons dégager trois principaux groupes de codes à quatre chiffres :

$$S = A.D + C.D + B.D$$

Simplifions par la loi distributive :

$$S = D.(A + B + C)$$

Le circuit logique prendra donc la forme :



Le prochain article fera la synthèse des notions de logique déjà vues, et proposera de nombreux exercices sur tous ces aspects.

Exercice 5

1° Simplifiez les expressions booléennes suivantes à l'aide de tableaux de Karnaugh :

a) $A.B.C + A.\bar{B}.\bar{C} + \bar{A}.\bar{C} + \bar{A}.B.C + A.B.\bar{C}$

b) $\bar{B} + \bar{C} + B.\bar{C} + A.C$

c) $A.\bar{B}.D + \bar{A}.D + A.B.C.D + A.B.\bar{C} + \bar{A}.B.\bar{C}.\bar{D}$

2° Vous devez créer un circuit logique qui rende compte de la représentation binaire des nombres entiers compris entre 0 et 7. Le circuit devra produire un résultat en sortie lorsque le nombre donné en entrée est impair ou multiple de 3 (par exemple 3 ou 6). En établissant une table de vérité, on pourra simplifier l'expression et dessiner le circuit logique répondant à cette fonction.

Réponses à l'exercice 4

1° a)

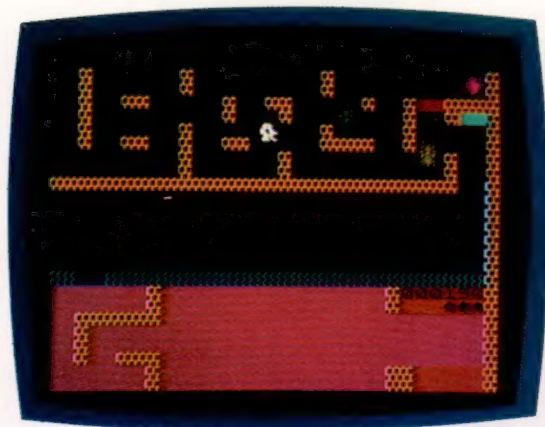
	A	\bar{A}	
B	0	1	
\bar{B}	0	1	
B	1	1	
\bar{B}	0	1	
	$\bar{B} + \bar{C}$		

b)

	A	\bar{A}	
B	0	0	
\bar{B}	0	0	
B	1	0	
\bar{B}	1	1	
	$(A + B).\bar{C}$		

Aventures

Être le héros pur et invincible des histoires les plus folles sur des planètes remplies de pièges et de « méchants » est un rêve pour beaucoup de jeunes et de moins jeunes. Rêvez...



STYX (Spectrum)

C'est un jeu d'aventure qui consiste en un combat contre les forces mauvaises pour atteindre « l'être obscur » et le détruire. Armé seulement d'un fusil laser, vous devez détruire les araignées qui vous guettent aux recoins d'un sombre labyrinthe, plonger dans la rivière Styx, où vous affronterez des plantes mortelles et de dangereux bancs de piranhas; puis, après l'avoir traversée, vous combattrez les esprits des morts qui essaieront de vous empêcher d'atteindre l'être obscur.



WACKY WAITERS (Vic-20)

Tout l'intérêt du jeu est centré sur l'habileté du joueur à atteindre l'autre côté de l'hôtel, où un client appelle, en utilisant une série d'ascenseurs en mouvement. Une fois arrivé de l'autre côté, le client vous demande d'aller lui chercher du vin et de le lui rapporter, en utilisant toujours les ascenseurs. Si vous vous faites remarquer par le directeur de l'hôtel, il vous donnera un avertissement écrit, et, au bout de trois fois, le jeu sera terminé. Si vous menez à bien votre tâche, vous recevrez un pourboire qui constitue le score du jeu.



EXTERMINATOR (CBM 64)

Vous devez sûrement vous rappeler le fameux CENTIPÈDE, un des meilleurs jeux d'arcades de ces dernières années. Eh bien, EXTERMINATOR est aussi attractif que CENTIPÈDE et même, peut-être, un peu plus. En effet, en plus des créatures que vous avez déjà rencontrées dans CENTIPÈDE, vous allez devoir vous affronter à un aigle fou, une énorme tarentule, des moustiques qui attaquent en piqué, et à douze douzaines de scorpions blancs ! Chaque créature se déplace de manière différente avec des effets sonores distincts. Un jeu extraordinaire !



VENUS VOYAGER (Atari)

C'est un mélange de jeu d'alunissage et de sauvetage d'humanoïdes. Vous disposez de sept vaisseaux et de plusieurs aires d'alunissage et vous devez piloter votre vaisseau jusqu'à une des aires pour recueillir les humanoïdes qui se trouvent à proximité, tout en évitant les murs des canyons lunaires, les mines volantes et une station orbitale qui pourra vous être utile si vous désirez vous ravitailler. Les graphiques et les effets sonores sont excellents, mais l'alunissage est très délicat et demande beaucoup d'expérience.



Ordinateur sans boîtier

Certains micro-ordinateurs sont livrés sans boîtier. Parmi ceux-ci, l'AIM-65 (*Advanced Interactive Microcomputer*) de Rockwell est conçu pour l'enseignement et le développement de programme.

L'AIM-65 est dépourvu de boîtier. Il est, malgré ce dépouillement, le seul ordinateur qui incorpore une imprimante. Sa conception est beaucoup plus complète que celle de la plupart des systèmes qui se proposent également d'être développés. Par rapport à ces derniers, il comporte en plus un clavier et un mode d'affichage à diodes témoins de vingt caractères et seize segments.

Il existe un autre modèle de ce micro-ordinateur AIM, avec clavier hexadécimal, affichage huit caractères et sept segments, et dépourvu d'imprimante. Mais il est beaucoup plus difficile à utiliser. Les caractéristiques distinctives de l'AIM-65 sont les canaux d'entrée-sortie et la mémoire. Cette dernière est, en standard, de 4 K de RAM et de 12 K de ROM. A cet égard également, l'AIM-65 est plus performant que certaines machines extensibles souvent dotées seulement de 1 K de RAM, et de 2 ou 4, rarement de 8 K de ROM.

En comparaison des ordinateurs individuels de gestion, qui ont presque tous 16 K de RAM

et souvent davantage, 4 K peuvent sembler dérisoires. Mais l'AIM-65 comporte en outre deux connecteurs pour l'adjonction de cartes mémoire supplémentaires, telle la carte de 32 K statiques de RAM.

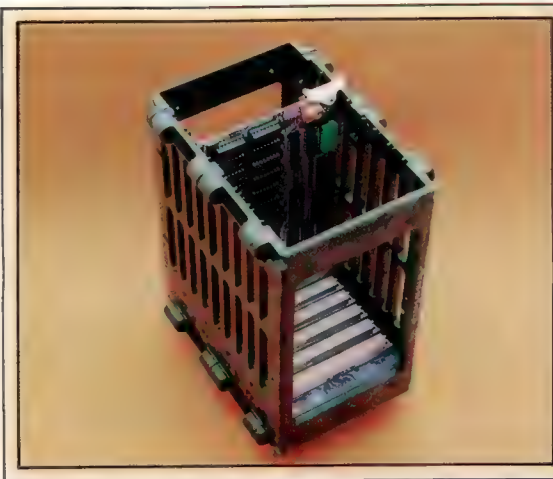
De manière générale, ces systèmes dits « de développement » sont utilisés pour des applications qui n'ont pas besoin de beaucoup de mémoire. Du reste, les ressources offertes par les 3 K disponibles après utilisation de la mémoire par le système sont très suffisantes, souvent à la grande surprise de l'utilisateur.

L'AIM-65 est construit autour d'un microprocesseur 6502, ce qui suppose que les zones mémoire possèdent des limites fixes. Les 512 octets de mémoire basse seront donc nécessairement réservés à la RAM. La « page zéro » est une zone spéciale comprise entre \$0000 et \$00FF, utilisée par le 6502 comme « pseudo-registre ». Les décodeurs d'adresse n'ont pas besoin d'adresse page (00000000 en binaire) dans les instructions. Celles-ci deviennent donc plus courtes et plus rapides à exécuter. Cette



AIM à prendre...

Le système complet AIM n'est pas d'un aspect très soigné, aussi le clavier et la console peuvent-ils être dotés de boîtiers. Cela convient parfaitement à l'usage standard avec 4 K de RAM. Pour les extensions mémoire, il est possible de connecter une carte directement à l'arrière. Si vous voulez connecter une autre carte, il vous faudra utiliser une boîte d'extension reliée à l'ordinateur par les deux cartes et la nappe figurant au premier plan. La boîte bleue en métal est l'alimentation. (Cl. Chris Stevens.)



Carte d'extension mémoire
Voici une boîte d'extension pouvant recevoir jusqu'à huit cartes (il existe également des boîtes pour quatre ou seize extensions). Les cartes sont de taille standard et se branchent par l'intermédiaire de leur connecteur standard Euro à quatre-vingt-seize broches. Vous pouvez agrandir la mémoire jusqu'à 128 K de ROM et de RAM. Une interface standard IEEE, ainsi que des cartes de connexion avec un lecteur, des moniteurs et une imprimante pleine page sont également prévus.

caractéristique contribue à la réussite du microprocesseur 6502, et confère au composant 256 des registres définissables par l'utilisateur.

La ROM de l'AIM est relativement importante et se trouve en mémoire haute, là où la situe le 6502. Divers langages et utilitaires sont possibles. Ils dépendent de la ROM moniteur, un bloc de 4 K comprenant des systèmes utilitaires de bas niveau (liés à l'ordinateur), situé entre \$E000 et \$EFFF. Ce bloc comprend un éditeur de lignes, l'exécution pas à pas (vérification et changement éventuel du contenu des positions mémoire), le mode trace (affiche le contenu du registre des positions mémoire pour chaque étape de l'exécution), ainsi que les modifications registre et mémoire.

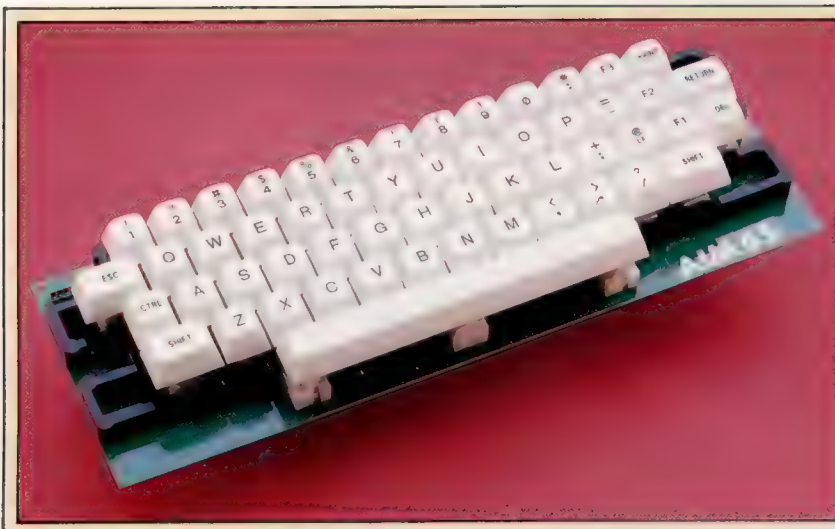
La machine est livrée avec cinq supports ROM pour branchement de logiciels câblés. Parmi ceux-ci, une version de BASIC comprend les principales fonctions, ainsi que la virgule flottante sur cinq octets et des fonctions arithmétiques définies sur une seule ligne.

Pour les acheteurs de l'AIM-65 intéressés par le contrôle des processus de fabrication industrielle ou par d'autres applications, les autres blocs de mémoire morte comprennent un

assembleur et un langage méconnu mais très intéressant, le PL/65, qui ressemble un peu à l'ALGOL ou au PL/1, et qui compile en un code source assembleur. Ce dernier peut ensuite être travaillé pour obtenir davantage de finesse et assemblé par l'assembleur.

Un autre langage, l'INSTANT PASCAL, est également disponible. Il s'agit d'une forme inhabituelle du PASCAL, interactif et interprété. Il présente à la fois la structure du PASCAL et la souplesse d'utilisation du BASIC. Le PASCAL lui-même, cependant, est trop important pour être résident dans l'AIM-65. Par contre, le FORTH est disponible avec pratiquement les mêmes avantages (il demande moins, de la part de la machine).

Du fait que les micro-ordinateurs de développement sont destinés aux ingénieurs (souvent sous forme de contrôleurs dans des machines complexes), leur technologie et leurs caractéristiques d'entrée-sortie sont plus développées et perfectionnées que sur la plupart des ordinateurs non spécialisés que l'on rencontre sur les marchés internationaux. Le micro-ordinateur AIM-65 dispose de nombreux canaux d'entrée-sortie, même sur le modèle de base.



Clavier
Le clavier de l'AIM permet la programmation en BASIC et en d'autres langages. De nombreux systèmes n'autorisent que des ensembles de touches en hexadécimal (0-9, A-F), et sont donc plus difficiles à programmer. PRINT en haut à droite est à utiliser conjointement avec CTRL pour activer ou inhiber l'imprimante incorporée. Quand cette dernière est activée, tout ce qui apparaît à l'écran est simultanément imprimé. (Cl. Chris Stevens.)

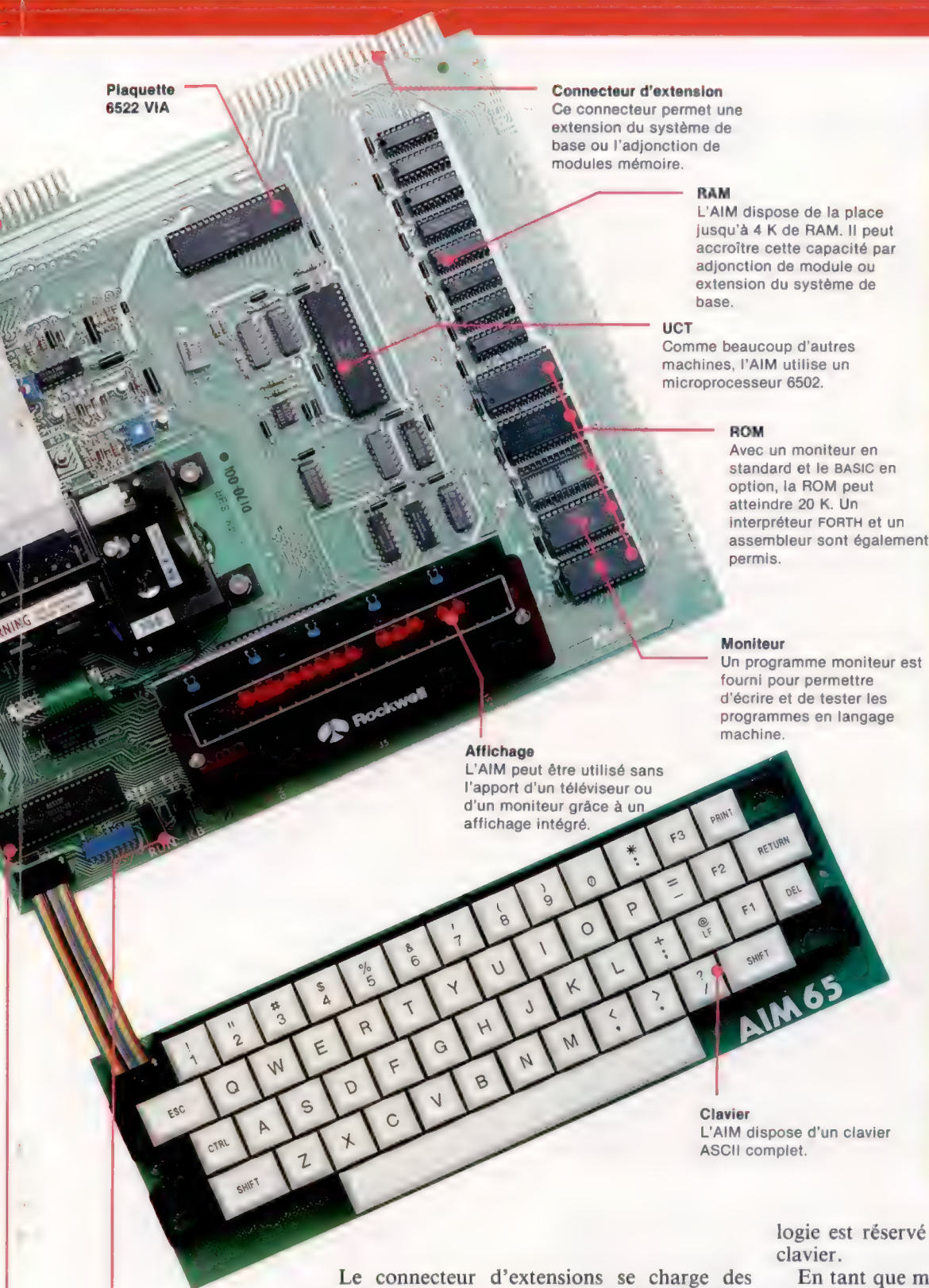
Connecteur d'application
Ce connecteur est muni de deux ports pour connecter l'AIM à tous les dispositifs qu'il doit contrôler.

Connecteur de puissance
L'AIM a besoin des 5 V classiques pour l'ordinateur et de 24 V pour conduire l'imprimante.

Imprimante
L'AIM permet, ce qui est rare, un contrôle permanent écrit des applications grâce à une imprimante thermique.

Bouton de remise à zéro

Décodeur clavier
Une puce 6532 RIOT contrôle le clavier.



Plaquelette 6522 VIA

Connecteur d'extension
Ce connecteur permet une extension du système de base ou l'adjonction de modules mémoire.

RAM
L'AIM dispose de la place jusqu'à 4 K de RAM. Il peut accroître cette capacité par adjonction de module ou extension du système de base.

UCT
Comme beaucoup d'autres machines, l'AIM utilise un microprocesseur 6502.

ROM
Avec un moniteur en standard et le BASIC en option, la ROM peut atteindre 20 K. Un interpréteur FORTH et un assembleur sont également permis.

Moniteur
Un programme moniteur est fourni pour permettre d'écrire et de tester les programmes en langage machine.

Affichage
L'AIM peut être utilisé sans l'apport d'un téléviseur ou d'un moniteur grâce à un affichage intégré.

Clavier
L'AIM dispose d'un clavier ASCII complet.

Interrupteur marche-saut
Permet à l'ordinateur un traitement normal ou de sauter une instruction.

Le connecteur d'extensions se charge des principaux signaux, dont on retiendra principalement les données et les lignes d'adresse, les signaux horloge et l'alimentation. Il existe en outre deux composants adaptateurs d'interfaces multiples 6522, dont l'un est destiné à contrôler l'imprimante, l'interface télétype et l'interface cassettes. Le deuxième composant n'est pas attribué et figure sur le connecteur d'application J2 sous la forme de deux ports bidirectionnels « huit lignes plus deux ». Un rythmeur 6532 RAM entrée-sortie est également fourni, mais ce dispositif complexe et de haute techno-

logie est réservé exclusivement à la gestion du clavier.

En tant que machine pour le développement, le micro AIM-65 n'est pas réservé au BASIC ou aux autres langages évolués. Il est conçu pour permettre aux utilisateurs de programmer leurs propres composants de ROM enfichables dans les supports ROM. La machine devient alors très spécialisée.

Dans l'ensemble, l'AIM-65 est un ordinateur d'une seule pièce à l'état brut, sans boîtier, mais modulaire et très bien conçu. Ses ressources s'adressent à quiconque cherche une machine compacte et très pratique, sans les caractéristiques des ordinateurs de gestion.

AIM-65

PRIX

DIMENSIONS

292 x 267 x 60 mm.

UCT

6502.

MÉMOIRE

ROM de 4 K avec supports pour adjonction de mémoire jusqu'à 20 K, et 4 K de RAM pouvant aller jusqu'à 64 K.

AFFICHAGE

Pas d'affichage vidéo, mais affichage à diodes témoins sur 20 caractères et 16 segments. Une carte contrôleur est nécessaire pour obtenir une sortie vidéo.

INTERFACES

Deux ports bidirectionnels sur 8 bits, chacun avec deux lignes de contrôle, plus le bus système (carte « fond de panier »).

LANGAGES

Mini-assembleur et éditeur de ligne fournis. Assembleur complet, BASIC, FORTH, PL/65 et INSTANT PASCAL disponibles.

CLAVIER

53 touches type machine à écrire, dont 3 touches de fonction.

DOCUMENTATION

Manuels d'installation et de présentation, véritables modèles de clarté et d'exhaustivité.

FORCES

Avantage principal : extrême modularité de la machine due à ses possibilités d'extension et au grand nombre de langages accessibles.

FAIBLESSES

Sans défaut majeur. Manque relatif cependant de logiciels évolués.

Faire des affaires

L'apparition de micro-ordinateurs bon marché et la difficulté de rédiger des programmes d'application bien spécifiques ont fait naître un marché du logiciel de gestion.

Le terme « logiciel de gestion » recouvre des applications très différentes. La première distinction est à établir entre programmes sur cassette et programmes sur disquette. Les premiers s'adressent essentiellement aux particuliers. Les disquettes, d'une capacité mémoire bien plus importante, plus rapides, où l'on peut lire et écrire des données bien plus aisément, devraient accueillir tous les logiciels proprement commerciaux; mais tout le monde n'est pas disposé à payer jusqu'à plusieurs milliers de francs pour cela.

Par ailleurs, il convient de distinguer les ordinateurs utilisés par une seule personne de ceux qui servent à nombre de gens, par l'intermédiaire de terminaux séparés. Nous commencerons par les systèmes les plus simples.

Le logiciel de gestion est davantage destiné au profane qu'à l'expert informaticien. C'est pourquoi, qu'il soit installé sur cassette ou sur dis-

quette (ou même sur disque dur), il aura presque toujours une forme « interactive ». Cela veut dire que tout au long des opérations l'utilisateur se verra guidé par une série de messages et d'injonctions qui s'afficheront à l'écran.

En règle générale, ces opérations prendront la forme d'options classées par ordre numérique au sein d'un « menu ». En choisir une mène parfois à un deuxième et même à un troisième menu. L'ordinateur réclame alors des données, par un message d'écran; l'information prend ordinairement une forme simple, semblable à celle d'un livre de comptes conventionnel. Cette facilité d'emploi a le plus contribué au succès de ce genre de programmes.

Nous considérerons trois domaines bien séparés : le livre des ventes, le livre des achats et le livre de comptes général (ou nominal). Le premier enregistre toutes les ventes effectuées par une entreprise au cours de ses activités courantes. Le deuxième relève tous les achats nécessaires à la poursuite de ces activités. Le troisième donne une image globale de la situation de l'entreprise, ainsi que l'état de son compte en banque, à un moment donné.

Les logiciels de gestion sont composés de fichiers, dont chacun abrite un certain nombre de dossiers, eux-mêmes divisés en zones. La distinction entre les trois est très simple. Un programme de ventes comprendra par exemple un fichier clients, contenant tous les dossiers de chacun d'eux; et les détails relatifs à chaque client, comme son nom et son adresse, seront enregistrés dans deux des zones du dossier correspondant.

Un progiciel doit, en outre, être capable de manipuler des données. Parmi les techniques de programmation indispensables à cet effet, il faut citer les routines d'entrée (pour communiquer les informations au système), ainsi que les routines de calcul, qui brassent des valeurs numériques contenues dans certaines zones, à l'intérieur des dossiers, ou de l'un à l'autre.

Les programmes généraux — installés sur disquette — mettent en œuvre des données très détaillées au niveau des ventes et des achats. Tous sont enregistrés (on dit « postés » en termes de comptabilité) par rapport à chaque client individuel et à chaque fournisseur. L'idée générale est de donner un tableau, aussi fidèle que possible, des transactions avec les uns et les autres.

Toutefois, même le micro-ordinateur le plus puissant, pourvu du programme le plus sophis-

L'argent, c'est du temps

Un bon moyen de surmonter les limitations des cassettes consiste à vendre des programmes séparés, qui peuvent être utilisés ensemble si cela se révèle nécessaire.

Un logiciel de gestion fonctionne parfaitement, mais vous devrez tenir vous-même votre comptabilité. Inversement, il peut utiliser des fichiers créés par d'autres logiciels, ce qui est un premier pas vers un système entièrement informatisé. Vous pouvez de cette façon mettre peu à peu toute votre gestion sur ordinateur, au lieu d'être contraint à un changement brutal.

ETAT du STOCK d'ARTICLES en MAGASIN au 01/01/80			
CODE	EDMS	ARTICLE	EDITEUR
QUANTITE MAGASIN	P. U. MOY. PERIODE	VAL. STOCK AU P. R.	
348	350.0000	137001.00	
QUANTITE VENDUE	P. U. MOY. PERIODE	CH. AFF. DE PERIODE	
4	850.0000	3400.00	
% REMISE MOYEN	P. V. CATALOGUE	MARGE BENEFICIAIRE	
0.00	850.00	451.50	

ACTIF	
ACTIFS COURANTS	
CHIQUE	000000.00
COMPTES CLIENTS	100000.00
STOCK	100000.00
ACTIF COURANT TOTAL	200000.00

Livre de comptes

Les progiciels installés sur cassette sont limités dans leur étendue et leur variété. C'est pourquoi un programme comme celui-ci se concentre sur une seule tâche. Il accomplit certaines fonctions comptables. (Doc. Goupil.)

tiqué, ne peut garder en mémoire qu'une certaine quantité de données; les possesseurs de cassettes sont encore moins bien lotis à cet égard. Si l'on envisage donc de faire l'achat d'un logiciel de gestion (sur cassette comme sur disquette), il faudra se poser la question : « Est-ce qu'il sera capable de traiter tous les éléments dont j'ai besoin pour mes affaires? »

Plus un fichier accueille de données, plus l'ordinateur a besoin de temps pour les trier, et la méthode habituelle consiste à ne conserver d'informations détaillées que pour les opérations en souffrance, ou « arriérés » (c'est-à-dire celles pour lesquelles l'argent est encore dû).

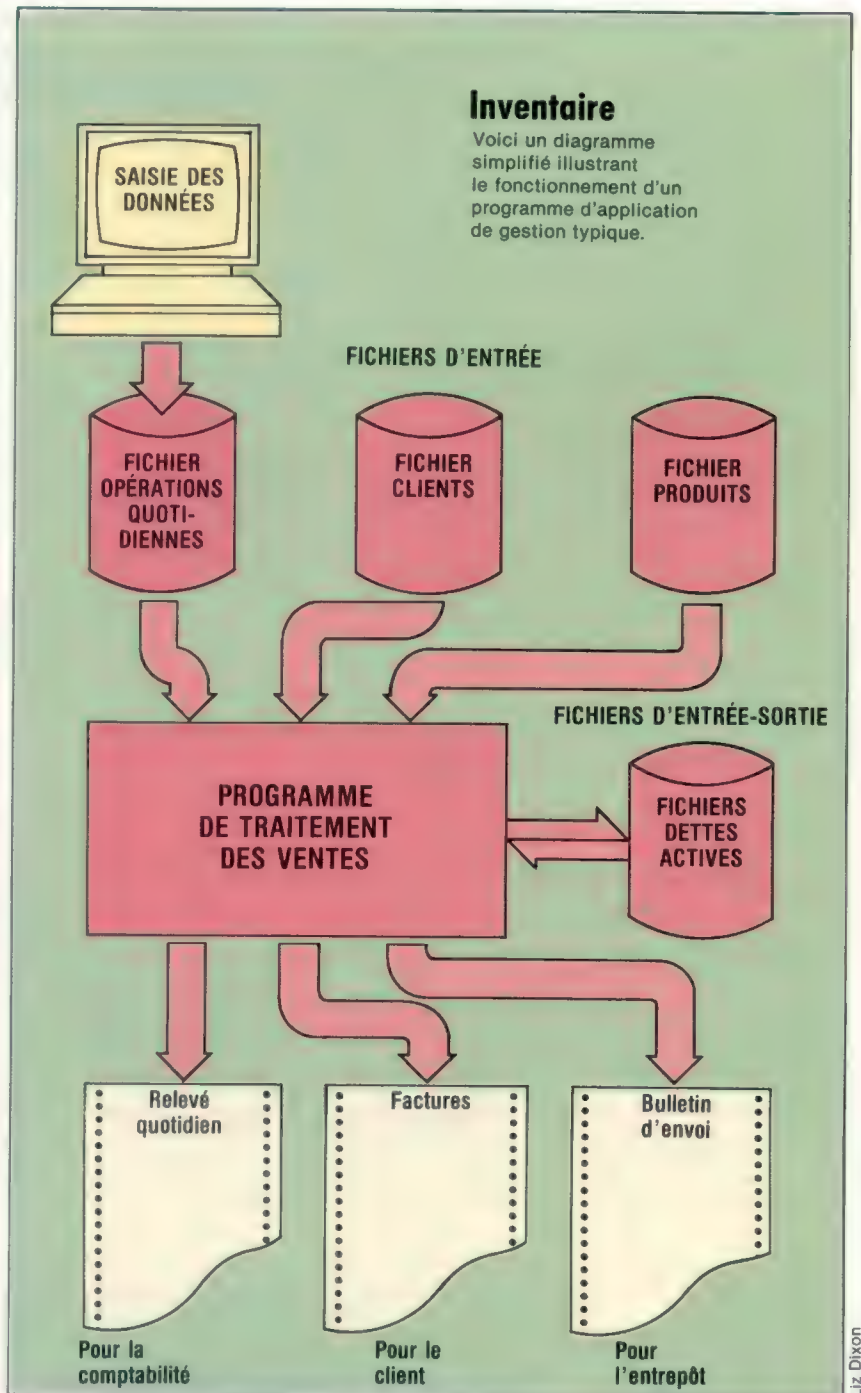
Un autre procédé consiste à ne retenir que la valeur cumulative des transactions entre l'entreprise, ses clients et ses fournisseurs. C'est ce qu'on appelle en comptabilité un report. Il véhicule moins d'informations, mais demande moins d'espace mémoire, et il est d'emploi plus facile. En d'autres termes, c'est un compromis entre l'efficacité (au niveau de la précision des détails) et les possibilités de calcul et de mémoire de l'appareil.

Comme nous le verrons plus tard, les programmes de ventes et d'achats installés sur disques ont des caractéristiques propres, qui ne peuvent être intégrées dans les logiciels sur cassette. C'est ainsi que, s'agissant des ventes, les premiers comportent une option facturation, qui permet l'envoi de factures et de relevés aux clients. En ce qui concerne les achats, il est de même possible d'imprimer des chèques ou des avis de versement (comportant la liste exacte de ce qui a été commandé) et de les faire parvenir aux fournisseurs.

Les logiciels de gestion ont de nombreuses fonctions. Le meilleur moyen de s'en faire une idée est de voir comment l'argent entre et sort. La première exigence d'un commerçant ou d'un homme d'affaires est de savoir quel est le revenu exact de son entreprise, et quel est le montant des dépenses. Le livre de comptes est, depuis longtemps, la solution la plus simple pour le savoir. Sous sa forme manuscrite, il comporte des pages divisées en autant de colonnes qu'il y a de sources de recettes et de dépenses. Il faut également tenir compte de la T.V.A. sur chaque opération, de façon à payer (ou à se faire rembourser) cette taxe, que ce soit par l'administration des douanes ou celle des contributions.

Enfin le livre de comptes doit être mis à jour, de façon quotidienne ou hebdomadaire. Chaque transaction peut être notée, bien que généralement on se borne à noter la valeur totale d'une journée. La différence entre ce type de registre et un logiciel de comptabilité générale tient au manque de précision du premier.

Les programmes installés sur cassette n'ont qu'une capacité mémoire limitée, parce qu'ils ne peuvent, comme les lecteurs de disquettes, enregistrer des données et vider une partie de leur mémoire pour leur faire de la place. C'est pourquoi ils reproduisent, parfois d'assez près, la structure d'un livre de comptes.



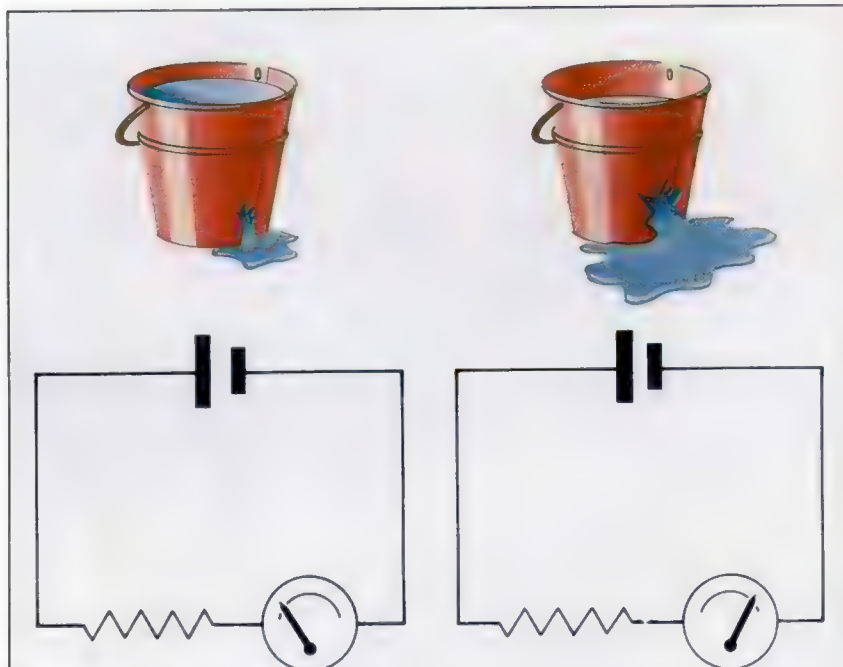
Au lieu de consacrer trois cassettes différentes (ventes, achats, comptabilité d'ensemble) à tous les mouvements de fonds, ils effectuent une sorte de synthèse.

Prenons un exemple typique, celui d'un petit commerçant effectuant le relevé de ses recettes et de ses dépenses, sur une base hebdomadaire. Un programme sur cassette rassemblera et additionnera toutes les données tapées d'une semaine à l'autre. Pour cela il fera la somme des recettes, puis la somme des dépenses, soustraira la seconde de la première, et indiquera ensuite les bénéfices ou les pertes.

La prochaine fois nous verrons à quelles exigences obéit la rédaction de logiciels de ce genre.



L'intendance

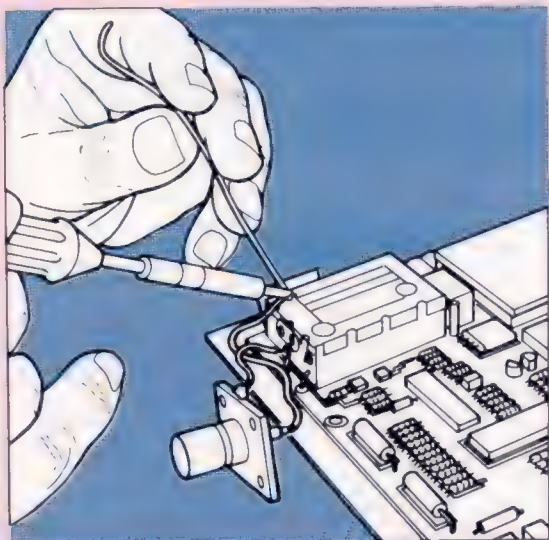


Résistance et courant

Un composant d'un circuit électrique oppose une résistance au courant, de la même manière que les tuyaux et les réservoirs d'un chauffage central opposent une résistance à la circulation de l'eau. Plus grande est la résistance, moindre est le courant. Si un seau comporte un petit trou, l'eau s'échappera lentement du fait des résistances. Lorsque le trou est plus grand, ces dernières diminuent et la fuite s'accroît. De manière similaire, un circuit dont la résistance est élevée présente un courant peu important. La résistance électrique dépend du matériau. Les fils électriques sont en cuivre et opposent peu de résistance. Les filaments des ampoules électriques présentent une résistance élevée, de sorte que le travail effectué par le courant contre la résistance s'exprime en chaleur, et donc en lumière.

Contrôle de la qualité

Le Spectrum de Sinclair peut fournir un signal vidéo composite, permettant l'utilisation d'un moniteur à haute résolution. Ouvrez le boîtier et repérez le modulateur RF — un gros composant argenté en haut à gauche. Il comporte deux branchements. Prenez celui qui est le plus près du dos. Soudez à partir de lui un court fil jusqu'au socle d'une prise BNC. Soudez un autre fil depuis la structure du modulateur lui-même jusqu'au connecteur central de la prise.



ATTENTION! La garantie d'un ordinateur peut devenir nulle si vous ouvrez le boîtier.

Le précédent article de cette série nous a montré le fonctionnement d'un système électrique. Nous abordons ici les trois composants électriques de base : le potentiel, le courant et la résistance. Regardons quelles relations existent entre eux pour en déduire l'équation mathématique qui les relie.

Georg Ohm, physicien allemand du XIX^e siècle, a été le premier à établir la relation entre tension, courant et résistance, et à en déduire que l'un de ces éléments pouvait toujours être déduit des deux autres. Il a ainsi découvert que la tension était égale à la résistance multipliée par le courant. Il a ensuite transposé l'équation, trouvant que la résistance pouvait être obtenue en divisant la tension par le courant, et le courant calculé en divisant la tension par la résistance. Si l'on prend les symboles usuels dans le domaine de la physique, V représente la tension, I le courant et R la résistance, voici les équations :

$$V = R \times I$$

$$R = V/I$$

$$I = V/R$$

La loi de la puissance est une extension de la loi d'Ohm :

$$W = V \times I$$

Elle indique que W, la puissance consommée dans un circuit ou dans un appareillage électrique, est proportionnelle à la tension et au courant qui traverse le circuit. Si la tension est exprimé en volts (V) et le courant en ampères (A), la puissance, W, sera alors mesurée dans une nouvelle unité, le watt.

Cette équation est d'une grande utilité; elle nous permet notamment de calculer la puissance nécessaire pour un fusible. Prenons l'exemple d'un ventilateur de 3 kW et de 240 V. En divisant 3000 (la puissance en watts consommée dans le circuit), par 240 (la tension), nous obtenons le courant électrique, soit 12,5 A. Le fusible devra être égal à 12,5 A. Du fait qu'une prise électrique ne peut accepter que 13 A, nous en concluons qu'il ne peut y avoir sur la même prise un autre appareil électrique consommant plus de 120 W, faute de quoi le circuit serait menacé. Nous donnerons dans le prochain numéro des exemples illustrant la mise en œuvre de la loi d'Ohm.



Les pièces d'un composant

Le diagramme d'un circuit électronique peut sembler rébarbatif *a priori*. Mais les diagrammes simplifient en fait les choses en décrivant de manière standard les

composants d'un circuit. Voici, pour illustrer cela, un circuit simple de communication. Sur un diagramme, chaque composant est représenté par un symbole et par un code tel que « R1 » ou « TR2 ». Cela se révèle utile pour se référer aux

composants, dans une liste de pièces par exemple. Les lignes reliant les composants sont mises pour des raisons de clarté. Ces lignes peuvent être de vrais fils ou des pistes sur une carte imprimée.

1. Commutateurs

Il existe de nombreuses variétés de commutateurs, chacune correspondant à une fonction déterminée. Les deux principaux types sont le commutateur à verrou où l'état du commutateur est maintenu après pression de la touche, et le commutateur sans verrou pour lequel le contact ne dure que le temps de la pression.



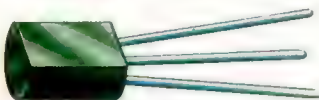
2. Diodes lumineuses

Les diodes sont le type le plus simple de semi-conducteurs. Ce sont les équivalents électroniques des valves sans retour qui permettent au courant de circuler dans un seul sens. Certaines diodes, encapsulées dans une résine translucide autre que le métal, restituent une certaine quantité de lumière et servent d'indicateurs.



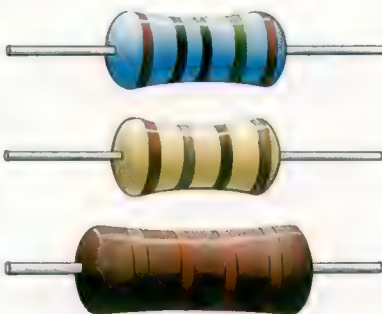
3. Transistors

Selon leurs spécifications et leur emploi, les transistors peuvent être des amplificateurs ou des commutateurs. Leur invention en 1947 a permis le développement de la micro-électronique.



4. Résistances

En introduisant des matériaux moins conducteurs dans un circuit, on peut contrôler le flux électrique. Les résistances peuvent être de tailles très diverses. Leur valeur est symbolisée par des bandes de couleurs.

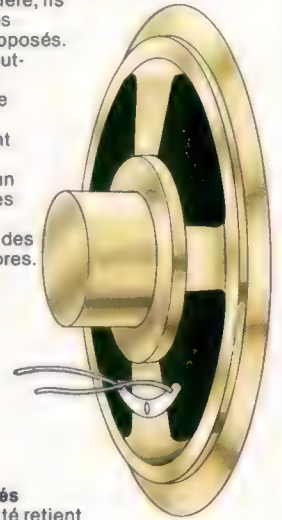


6. Piles

La plupart des petits circuits semblables à celui-ci peuvent recevoir comme alimentation de simples piles électriques, du fait qu'elles fournissent un courant direct et stable.

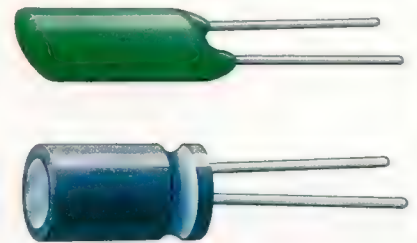
7. Haut-parleur

Les haut-parleurs ressemblent aux microphones : fonctionnant de la même manière, ils donnent des résultats opposés. Dans un haut-parleur, un diaphragme relié à un aimant vient vibrer en réponse à un courant. Ces vibrations produisent des ondes sonores.



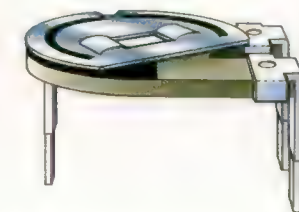
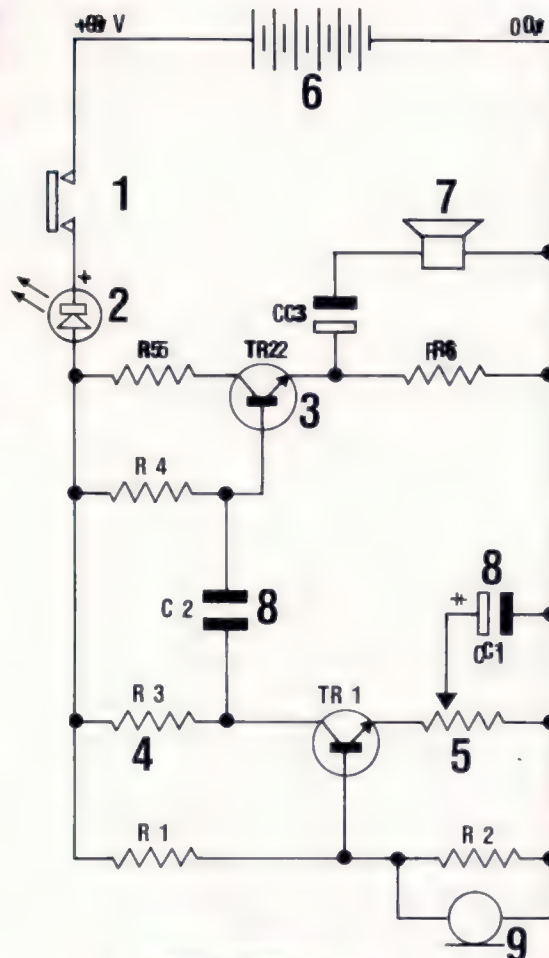
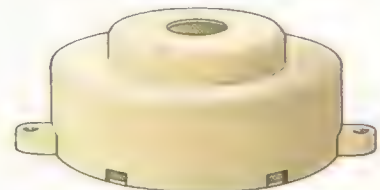
8. Capacités

Une capacité retient une charge électrique. Elle se charge lorsque ses deux fiches sont reliées au courant. Une fois chargée, plus rien ne change, même si le courant est interrompu. La capacité se déchargera lorsque ses deux fiches seront connectées entre elles.



9. Microphones

Les microphones fonctionnent à l'inverse des haut-parleurs. Les ondes sonores font vibrer un diaphragme qui génère, par l'intermédiaire d'un aimant, différents voltages dans le circuit.



5. Résistances variables

Les résistances variables, ou potentiomètres, utilisent un filament de carbone comme conducteur. La distance que le courant parcourt dans le filament avant d'atteindre l'extrémité du composant détermine la résistance.

Octet par octet

Nous suivons toute la procédure de développement d'un programme, depuis la définition de la tâche initiale jusqu'au langage machine lui-même en passant par l'interprétation en langage d'assemblage.

Dans les numéros précédents, nous avons vu comment des lignes de programme BASIC sont réduites en entrée sous forme d'octets de mots clés suivis de données en ASCII. Cela nous a montré que, bien que le BASIC soit certainement un langage évolué, il ne l'est pas tant que ça : il consiste essentiellement en une séquence d'instructions dont chacune consiste en une commande (aussitôt remplacée par un octet qui, lui-même, n'est que d'un niveau au-dessus d'un `opc` [code opération] machine) suivie des données correspondant à cette commande. Le fait que les commandes et leurs données (variables, nombres ou chaînes) soient plus proches du langage naturel, et que les instructions soient visiblement séparées par des numéros de lignes ou de colonnes, a pour effet qu'un programme BASIC nous semble bien plus évolué qu'il n'apparaît à l'interpréteur BASIC. Il s'ensuit qu'un langage machine ne devrait avoir besoin que d'être légèrement maquillé pour devenir raisonnablement intelligible à nos yeux.

Le « maquillage » du code machine que nous utilisons est le langage d'assemblage, dans lequel des mnémoniques alphabétiques comme LDA et ADC remplacent les octets `opc` que comprend en fait le microprocesseur, et où des symboles alphanumériques tels que LABEL1 et TFLAG peuvent être utilisés au lieu d'adresses mémoire et de données numériques. Le microprocesseur ne comprend pas le langage d'assemblage; aussi, avant qu'un programme puisse être exécuté, doit-il être traduit en langage machine — soit par un programme appelé assembleur, soit manuellement par le programmeur. La caractéristique du langage d'assemblage est qu'il devient du langage machine par traduction. En substituant simplement des `opc` aux mnémoniques, et des nombres aux symboles, il est transformé en un code exécutable. Mais il est bien plus compréhensible que le langage machine, c'est pourquoi il est extrêmement utile pour le développement de programmes. Nous écrirons toujours les programmes en langage d'assemblage, sans trop nous préoccuper de l'équivalent en langage machine jusqu'aux tout derniers stades de développement. Mais, pour le moment, nous ferons les deux pour des raisons d'intérêt et de clarté, sans oublier qu'en général le langage d'assemblage fera tout ce que nous voudrions.

Le microprocesseur peut accomplir de nombreuses opérations différentes, mais la seule chose qu'il fasse en réalité est de manipuler les

contenus de mémoire. Il le fait en agissant directement sur la mémoire de l'ordinateur — les puces RAM et ROM — ou bien sur sa propre mémoire interne qui est constituée en *registres*. Ceux-ci sont formés d'octets de mémoire physiquement localisés à l'intérieur de la puce de microprocesseur (ils ont certaines fonctions spéciales) mais qui sont indiscernables de tous les autres octets de mémoire.

Registre accumulateur

Le plus important des registres du microprocesseur est appelé *accumulateur*. Il est directement connecté à l'unité arithmétique et logique; il est donc utilisé plus souvent que les autres. Pour s'en servir, il faut pouvoir y mettre de l'information, processus appelé « chargement de l'accumulateur ». En langage d'assemblage, nous disons que le 6502 fait cela en accomplissant l'opération LDA, et dans le Z80 par l'opération LD A. Prendre de l'information de l'accumulateur est aussi essentiel que de charger celui-ci; en langage d'assemblage 6502 cela se fait par l'opération STA (STocke les contenus de l'Accumulateur). Le Z80, en revanche, considère le chargement et le stockage comme différents éléments de la même chose — à savoir un transfert de données. Ainsi, prendre de l'information du registre accumulateur est également fait par l'opération LD A, mais dans un format différent — comme nous le verrons par la suite.

Supposons que nous voulions écrire un programme en langage d'assemblage pour copier le contenu d'un octet de mémoire dans l'octet de mémoire suivant. Commençons par copier l'octet \$09FF dans l'octet \$0A00. Nous pouvons immédiatement exprimer cela en langage d'assemblage :

6502	Z80
LDA \$09FF	LD A,\$09FF
STA \$0A00	LD (\$0A00),A

Notez que nous copions le contenu de l'octet \$09FF dans l'octet \$0A00, sans savoir quel est ce contenu : il est vital de préciser cela d'emblée. L'octet \$09FF peut contenir tout nombre de \$00 à \$FF, et tout ce que fait notre programme est de charger ce nombre dans l'accumulateur, puis de le transférer de l'accumulateur dans l'octet \$0A00. La version 6502 de langage d'assemblage ne précise pas que LDA se rapporte au *contenu* de \$09FF, mais il fait une distinction sans équivoque entre le chargement (LDA) et le stockage (STA). La



version Z80 ne fait pas cette dernière distinction dans ses opc, mais son format d'instruction est toujours :

OPC DESTINATION, (SOURCE)

Cette version met des parenthèses quand elle veut préciser le « contenu de », ce qui renforce la distinction vitale entre l'adresse d'un octet et ce qu'il contient.

Le programme en langage d'assemblage que nous avons donné est logiquement complet, mais nous l'exécuterons comme un sous-programme, ce qui nécessite l'équivalent de la commande RETURN (pour retourner au programme principal). Les opc sont RTS en 6502, et RET en Z80.

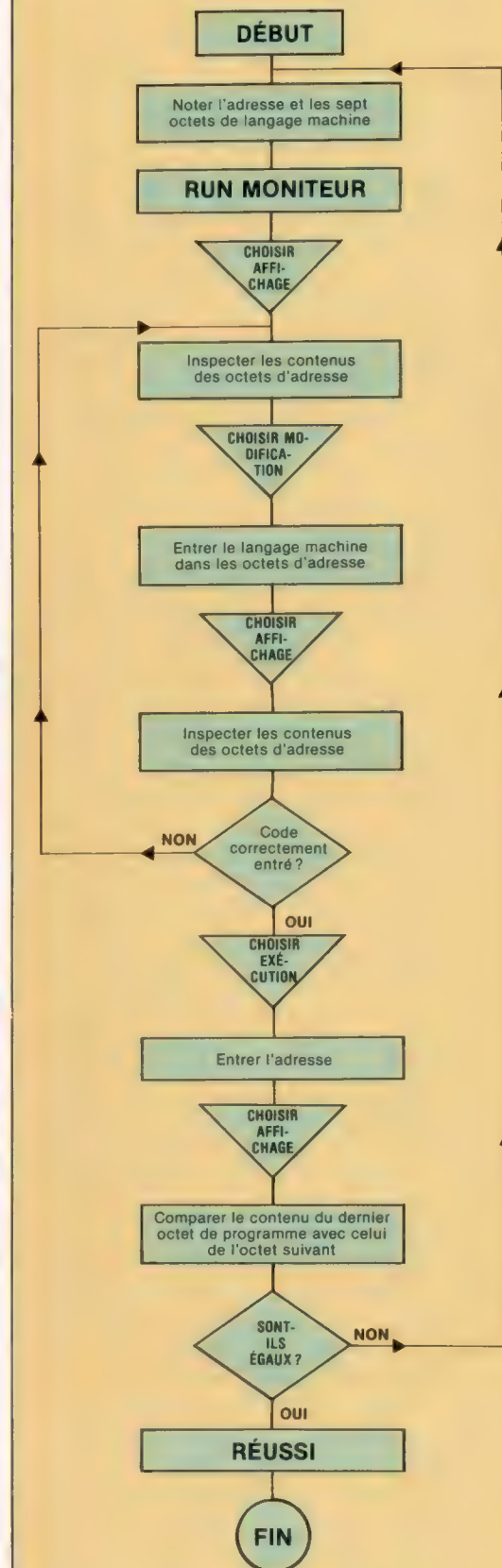
Si nous voulons utiliser ce sous-programme, il nous faut d'abord le traduire en langage machine, puis stocker le code quelque part en mémoire, et faire en sorte que le microprocesseur l'exécute. Nous pouvons utiliser le programme Moniteur pour ces deux dernières tâches; mais auparavant, nous devons faire la traduction et décider où doit aller le code.

Un programme en langage machine peut être stocké et exécuté n'importe où dans la mémoire, mais certains emplacements sont meilleurs que d'autres :

6502		
Adresse	Langage machine	Langage d'assemblage
COMMODORE 64		
\$0350	AD 56 03	LDA \$0356
\$0353	8D 57 03	STA \$0357
\$0356	60	RTS
BBC MICRO		
\$0070	AD 76 00	LDA \$0076
\$0073	8D 77 00	STA \$0077
\$0076	60	RTS
Z80		
SPECTRUM 16 K		
\$7FA0	3A A6 7F	LD A,(\$7FA6)
\$7FA3	32 A7 7F	LD (\$7FA7),A
\$7FA6	C9	RET
SPECTRUM 48 K		
\$FFA0	3A A6 FF	LD A,(\$FFA6)
\$FFA3	32 A7 FF	LD (\$FFA7),A
\$FFA6	C9	RET

Remarquez que chaque version du programme copie son dernier octet dans l'octet suivant. Le programme pour Spectrum 48 K, par exemple, copie le contenu de \$FFA6 dans \$FFA7. Notez également que les adresses apparaissent sous la forme hi-lo en langage d'assemblage (heureusement pour nous), mais en lo-hi dans la version en langage machine. En Z80, le mnémotique est LD dans les deux premières instructions, mais les opc diffèrent : 3A pour le transfert de données vers l'accumulateur, et 32 pour le transfert à partir de l'accumulateur.

Utilisation du programme Moniteur



Le programme de la page suivante vous permet de **MODIFIER**, **AFFICHER** et **EXÉCUTER** la mémoire. Chaque fois que vous choisissez une de ces options, vous devrez fournir une adresse hex. C'est l'adresse mémoire par laquelle commence : 1° soit la modification des contenus de mémoire; 2° soit l'affichage des contenus de mémoire; 3° soit l'exécution par le microprocesseur du programme en langage machine.

L'entrée de X au lieu d'un nombre ou d'une adresse vous ramènera toujours au niveau commande, et à ce niveau Q terminera l'exécution du programme.

Si vous sélectionnez **MODIFICATION**, lorsque vous avez donné l'adresse dont vous voulez modifier le contenu, cette adresse s'affiche, suivie d'une question; tapez le nouveau contenu hex et appuyez sur **RETURN**. L'adresse de l'octet suivant s'affichera, et vous répétez l'opération tant que vous voulez modifier des octets.

Chargement en mémoire
Dans le programme correspondant à votre machine, notez la première adresse et les sept octets de langage machine (\$0350 et AD, 56, 03, 8D, 57, 03, 60 sur le Commodore 64, par exemple); vous allez utiliser le programme Moniteur pour charger ces sept nombres hex dans les sept octets de mémoire à partir de la dernière adresse.
1° Faites tourner (**RUN**) le programme Moniteur, **AFFICHER** les contenus de mémoire à partir de la première adresse (de \$0350 à \$0357 pour le Commodore 64).
2° Choisissez **MODIFICATION**, entrez l'adresse et les sept octets de langage machine.
3° Choisissez **AFFICHAGE**, et assurez-vous que vous avez correctement entré les octets de langage machine dans les octets d'adresse.
4° Choisissez **EXÉCUTION**, et entrez l'adresse — rien ne doit se passer.
5° Choisissez **AFFICHAGE**, inspectez les adresses, et vous devriez trouver que les contenus des derniers octets du programme ont été copiés dans l'octet suivant.



Spectrum Monitor Program

```

48 REM+++++
49 REM+ +
50 REM+ HCAC MONITOR 1 +
51 REM+ ---- SPECTRUM ---- +
52 REM+ SAUVEGARDEZ CE +
53 REM+ PROGRAMME AVANT DE +
54 REM+ FAIRE 'RUN' +
55 REM+++++
100 GOSUB 1000 : REM + INIT+
200 CLS
300 PRINT " ** COMMANDES DE
HCAC MONITOR 1 ** "
400 FOR P = 1 TO LT:PRINT O$(P)
:NEXT P
500 FOR Z = 0 TO 1 STEP 0
550 GOSUB 2000 : REM + INPUT +
600 GOSUB (4500 + CH*500)
650 NEXT Z
700 STOP
750 REM+++ FIN DU PROGR.PRINC.
1000 REM *** S - P INIT *****
1050 LET LT = 4:DIM C$(LT):DIM
O$(LT,24):DIM X$(16)
1100 LET X$ = "0123456789ABCDEF"
1150 LET C$ = "MAEQ":LET C1 = - 48:
LET C2 = 10 - CODE(C$(1))
1200 LET O$(1) = " M - MODIFIER
MEMOIRE"
1220 LET O$(2) = " A - AFFICHER
MEMOIRE"
1240 LET O$(3) = " E - EXECUTER
LANGAGE MACHINE"
1260 LET O$(4) = " Q - QUITTER LE
PROGRAMME"
1300 RETURN
2000 REM *** S - P INPUT *****
2100 FOR P = 0 TO 1 STEP 0
2150 PRINT:PRINT "COMMANDE ?? "
2190 IF INKEY$ < > " " THEN GOTO 2190
2200 LET A$ = INKEY$:IF A$ = " " THEN
GOTO 2200
2250 FOR J = 1 TO LT
2300 IF A$ = C$(J) THEN LET CM = J:LET
J = LT:LET P = 2
2350 NEXT J: NEXT P:IF A$ = "Q" THEN
RETURN
2400 PRINT O$(CM)
2450 FOR P = 0 TO 1 STEP 0
2500 INPUT "ADRESSE HEX (X = QUITTER) "
:A$
2550 GOSUB 5200 :REM VERIF ET AJUST
2600 NEXT P:IF A$ = "X" THEN LET CM = 0
2650 RETURN
3000 REM *** S - P OCTET HEX *****
3010 LET HB = INT(N/16):LET LB = N - HB*16
3020 LET B$ = X$(HB + 1) + X$(LB + 1)
3030 RETURN
3100 REM *** S - P D - H *****
3110 IF NM < 256 THEN LET N = NM:GOSUB 3000
:LET H$ = "00" + B$:RETURN
3120 LET HI = INT(NM/256):LET LO = NM - 256*HI
3130 LET N = HI:GOSUB 3000:LET H$ = B$
3140 LET N = LO:GOSUB 3000:LET H$ = H$ + B$
3150 RETURN
4000 REM ***** S - P H - D ,
4050 LET RX = 1: LET DN = 0:LET HL = LEN(H$):
IF (HL < 1) OR (HL > 4) THEN
LET DN = - 1:RETURN

```

```

4100 FOR H = HL TO 1 STEP - 1
4150 LET D$ = H$(H)
4200 LET V = CODE(D$) + C1*(D$ > = "0"
AND D$ < = "9") + C2*(D$ > = "A" AND
D$ < = "F")
4250 IF V > 15 THEN LET DN = - 1:LET
H = 1:NEXT H:RETURN
4300 LET DN = DN + V*RX:
LET RX = RX*16
4350 NEXT H:RETURN
4500 REM **** S - P FACTICE *****
4550 RETURN
5000 REM **** S - P MODIF *****
5020 FOR P = 0 TO 1 STEP 0
5040 PRINT A$:INPUT "NOUVELLE
VALEUR HEX (X = QUITTER) ? " :V$
5050 PRINT V$
5060 GOSUB 5340 :REM VERIF ET OBEIT
5080 NEXT P:RETURN
5200 REM ** S - P VERIF ET AJUST ***
5220 IF A$ = "X" THEN LET P = 2:RETURN
5240 LET LL = LEN(A$):IF LL > 4 THEN
RETURN
5260 LET H$ = A$:GOSUB 4000
5280 IF DN > = 0 THEN LET P = 2:LET NM
= DN
5300 LET A$ = A$ + " " :IF LL < 4 THEN
LET A$ = "0000"(TO 4 - LL) + A$
5320 RETURN
5340 REM ** S - P VERIF ET OBEIT ***
5360 IF V$ = "X" THEN LET P = 2:RETURN
5380 LET H$ = V$:GOSUB 4000
5400 IF (DN < 0) OR (DN > 255) THEN
RETURN
5420 POKE NM, DN
5440 LET NM = NM + 1:IF NM > 65535 THEN
LET P = 2:RETURN
5460 GOSUB 3100 :REM S - P D - H
5480 LET A$ = H$ + " " :RETURN
5500 REM **** S - P AFFICH *****
5520 FOR P = 0 TO 1 STEP 0
5540 INPUT "NO.DEC. D'OCTETS
(X = QUITTER) ":N$:IF N$ = "X" THEN LET
P = 2:NEXT P:RETURN
5560 LET BN = VAL(N$):IF (BN > 0) AND
(BN + NM < 65536) THEN LET P = 2
5580 NEXT P
5600 FOR B = NM TO (NM + BN - 1)
STEP 4
5620 LET L$ = " " :LET NM = B:GOSUB 3100
5640 PRINT H$:TAB(6):
5660 FOR C = 0 TO 3
5680 LET N = PEEK(B + C):LET K$ = "."
5700 GOSUB 3000 :REM S - P D - H
5720 PRINT TAB(6 + 4*C):B$:
5740 IF N = 0 THEN LET K$ = " "
5760 IF (N > 31) AND (N < 128) THEN
LET K$ = CHR$(N)
5780 LET L$ = L$ + K$
5800 NEXT C
5820 PRINT TAB(26):L$
5840 NEXT B:RETURN
6000 REM***S - P EXECUT *****
6250 RANDOMIZE USR(NM):RETURN
6500 REM***S - P QUITTE *****
6550 PRINT TAB(5):"■■■■ FIN DE
PROGRAMME■■■■"
6600 LET Z = 2:RETURN

```

BBC Micro

```

39 REM*****
40 REM+ HCAC MONITOR 1 +
41 REM+ ----- BBC ----- +
42 REM+ MODIFIEZ LA VERSION +
43 REM+ DU SPECTRUM COMME +
44 REM+ SUITE : +
45 REM+ REMPLACEZ CODE: PAR +
46 REM+ ABC +
47 REM+ AJOUTEZ,MODIFIEZ OU +
48 REM+ EFFACEZ COMME INDIQUE +
49 REM+ +
50 REM*****
60 +TV 255
70 MODE 7
200 PRINT CHR$(147);CHR$(142)
600 ON CH GOSUB 5000,5500,6000,6500
1000 LT = 4:DIM C$(LT):D$(LT)
1150 C$(1) = "M":C$(2) = "A":C$(3) = "E":
C$(4) = "0":C1 = 48:C2 = ASC(C$(1)) - 10
2190 -----EFFACER-----
3020 B$ = MID$(X$,HB + 1,1) + MID$(X$,LB + 1,1)
4150 D$ = MID$(H$,H,1)
4500 -----EFFACER-----
4550 -----EFFACER-----
5050 -----EFFACER-----
5300 A$ = A$ + " " :IF LL < 4 THEN A$
= LEFT$("0000",4 - LL) + A$
5420 ?(NM) = DN
5600 N = 7*(B + C):K$ = "."
5740 IF N = 13 THEN K$ = CHR$(255)
6050 CALL NM:RETURN
6600 Z = 1:RETURN

```

Commodore 64

```

49 REM*****
50 REM+ HCAC MONITOR 1 +
51 REM+ ----- CBM ----- +
52 REM+ MODIFIEZ LA VERSION DU +
53 REM+ SPECTRUM COMME SUITE : +
54 REM+ +
55 REM+ REMPLACEZ TOUS LES : +
56 REM+ "LET P = 2" PAR "P = 1" +
57 REM+ +
58 REM+ REMPLACEZ CODE: PAR +
59 REM+ ABC: +
60 REM+ ET MODIFIEZ OU EFFACEZ +
61 REM+ COMME INDIQUE +
62 REM+ +
63 REM*****
700 PRINT CHR$(147);CHR$(142)
600 ON CH GOSUB 5000,5500,6000,6500
1000 LT = 4:DIM C$(LT):D$(LT)
1150 C$(1) = "M":C$(2) = "A":C$(3) = "E":
C$(4) = "0":C1 = 48:C2 = ASC(C$(1)) - 10
2190 -----EFFACER-----
3020 B$ = MID$(X$,HB + 1,1) + MID$(X$,LB + 1,1)
4150 D$ = MID$(H$,H,1)
4500 -----EFFACER-----
4550 -----EFFACER-----
5050 -----EFFACER-----
5300 A$ = A$ + " " :IF LL < 4 THEN A$
= LEFT$("0000",4 - LL) + A$
5740 IF N = 0 THEN K$ = " "
6050 SYS(NM):RETURN
6600 Z = 1:RETURN

```

Ce programme vous permettra d'afficher les contenus de mémoire, de les modifier et d'exécuter un programme stocké en langage machine.



Seul maître à bord

Jack Tramiel, grand patron de Commodore, a passé les rênes à Irving Gould. Il quitte une compagnie florissante. Comment a-t-elle pu arriver à des résultats qui font pâlir d'envie la concurrence?

Le secret de la réussite de Commodore tient avant tout à ses méthodes de production, organisée à l'échelon international. En Grande-Bretagne la firme a ainsi monté, dans l'ancienne ville sidérurgique de Corby, une usine très moderne, avec l'assistance financière du gouvernement britannique : elle devrait s'agrandir prochainement. Elle emploie deux cent cinquante personnes et produit cinq mille ordinateurs par jour. Comme toutes les usines de Commodore, elle bénéficie d'un approvisionnement assuré en semi-conducteurs : ils sont produits par la maison mère elle-même. Commodore est une organisation énorme et peut, au besoin, imposer des conditions draconiennes aux autres fournisseurs : dans certains cas la firme ne paie que la moitié du prix demandé aux autres.

Une position aussi solide doit beaucoup au succès du CBM PET (*Personal Electronic Transactor*, « opérateur électronique personnel »). L'appareil était dû avant tout à Chuck Peddle, qui prit à cette occasion trois décisions importantes. La première était de construire la machine autour du microprocesseur 6502; la deuxième était d'y implanter le BASIC Microsoft; la troisième était de le doter d'un véritable éditeur d'écran, bien plus maniable que les simples claviers manipulés avec passion par les premiers amateurs depuis l'apparition des premiers microprocesseurs vers le milieu des années soixante-dix. Aujourd'hui encore, l'IBM PC, ordinateur « de nouvelle génération », est dépourvu, dans sa version de base, de ce dispositif...

Commodore était bien placé pour fabriquer un tel appareil, puisque MOS Technology, détentrice des brevets du 6502, lui appartenait. C'était là une position exceptionnelle : les deux concurrents du PET, le TRS-80 et l'Apple II, faisaient usage de ce microprocesseur, ce qui permettait de leur tenir la dragée haute. Mais la naissance du PET n'eut pas lieu sans difficultés. Contre l'avis de Peddle, Tramiel insista pour que les composants mémoire soient aussi fournis par MOS Technology; il s'ensuivit un vif affrontement entre les deux hommes, puis le départ de Peddle.

Le PET est aujourd'hui un ancêtre vénérable. D'abord logé dans un boîtier d'acier embouti (comme les meubles de bureau), il a ultérieurement été relogé dans un ensemble en plastique plus moderne. La version la plus récente est celle de la série 8000, qui peut accéder à une

mémoire de 22 mégaoctets sur disque dur. En dépit de son âge, l'appareil se vend encore remarquablement bien aux consommateurs prudents, qui en apprécient la facilité d'emploi et ne voient pas de raisons de passer aux ordinateurs de bureau construits autour du 8088 d'Intel. Les fournisseurs déclarent même, à leur grand étonnement, qu'il est toujours compétitif face à l'IBM PC.

Il y a là une certaine part de hasard. Commodore, Apple et Tandy furent les premières firmes à proposer un ordinateur personnel d'emploi aisé et de prix raisonnable. Mais, une fois que Commodore s'est trouvé un client, il ne le lâche plus. A la fois par conservatisme et par désir de limiter les prix de revient, la compagnie ne s'est jamais souciée d'accroître les possibilités de ce qui n'était qu'un appareil pour amateur. Aussi, bien des programmes du PET peuvent-ils tourner sur les machines actuelles, tandis que le BASIC 2.0 de Microsoft reste très largement ce qu'il était du temps de Chuck Peddle.

Malheureusement pour ceux qui veulent aller plus loin, le 2.0 de Microsoft fut mis au point à une époque où le son et le graphisme étaient un luxe inabordable pour un ordinateur bon marché. Le Vic-20 et surtout le Commodore 64 en sont désormais très bien équipés, mais il leur manque des commandes spécifiques, ce qui impose de fastidieuses séries de POKE dans les adresses mémoire. Il est vrai qu'on peut toujours recourir à des programmes spécialisés installés sur cartouche. Commodore en propose un grand nombre.

C'est en partie grâce à un approvisionnement en composants assuré que la firme de Jack Tramiel est sortie victorieuse de la guerre des prix qui fut fatale à ses concurrents. Texas et Mattel se retirèrent du marché des micro-ordinateurs, et Atari est toujours en convalescence depuis un an ou deux. Commodore pouvait annoncer des baisses importantes sur les matériels et les logiciels et faire encore des bénéfices : il fut un temps où ses cartouches étaient vendues le tiers du prix moyen de celui de ses adversaires. Une production ultramoderne, automatisée, des composants peu coûteux, tout cela était le résultat de vingt ans d'expérience. On dit que le coût de production d'un Commodore 64 est inférieur à 500 francs...

Cette position dominante ne doit pas grand-chose à la chance. Commodore a connu des moments difficiles et a été deux fois au bord de



Jack Tramiel

Jack Tramiel, longtemps président de la compagnie, en a toujours été l'âme. Sa redoutable compétence commerciale va manquer à Commodore, maintenant qu'il n'est plus que « conseiller à temps partiel ». (Cl. Microscope.)



Chuck Peddle

Chuck Peddle a conçu à la fois le microprocesseur 6502 et le PET qui l'abritait. Il partit ensuite fonder sa propre compagnie, qui produisit le Sirius, un 16-bit orienté vers la gestion. (Cl. VNU.)



la faillite. Jack Tramiel, jeune adolescent polonais rescapé d'Auschwitz, s'installa aux États-Unis après la Seconde Guerre mondiale et fonda CBM — Commodore Business Machines — en 1955. La firme, installée à Toronto, au Canada, commença modestement par fabriquer des machines à écrire sous licence tchèque. On dit que Tramiel choisit un nom évoquant le plus possible celui d'IBM.

En 1975, après deux décennies consacrées à la vente d'équipement de bureau, Commodore faillit succomber lors de la féroce guerre des calculatrices, que les Japonais finirent par remporter. Mais Tramiel, que ses méthodes de gestion font à la fois craindre et respecter, ne s'avoua pas vaincu. Il se rendit compte que le marché de l'ordinateur personnel existait, engagea Chuck Peddle en 1976, et en moins de dix ans multiplia par cinquante la valeur financière de sa firme.

Commodore, malgré sa puissance industrielle et commerciale, a une grosse faiblesse : la recherche. Sa philosophie a toujours été : « Nous vendons aux masses, pas aux classes », et Tramiel pensait que le client achèterait forcément ce qui offrait le meilleur rapport

qualité/prix. Mais produire aux prix les plus bas possibles n'est peut-être pas un moyen très sûr d'intégrer les derniers progrès de la technologie. A la fin de 1982, plusieurs membres de la petite équipe de chercheurs de la firme la quittèrent en bloc, et, depuis, elle s'efforce de racheter des brevets aux autres. Elle a ainsi conclu des contrats en Extrême-Orient pour des lecteurs de disquettes, a engagé des pourparlers avec des compagnies comme Sony pour se procurer de coûteuses technologies « de cinquième génération », comme un système de reconnaissance de parole, des mémoires sophistiquées ou des robots domestiques.

En 1984, Commodore reste plus confiant que jamais, et continue à faire de la simplicité et du bon marché ses principaux arguments. La firme a présenté récemment deux nouveaux appareils construits autour d'un nouveau microprocesseur, le 7501 : il s'agit du 264 et du V-364, ce dernier étant équipé d'un synthétiseur de parole, avec un vocabulaire de deux cent cinquante mots. Conformément à la tendance actuelle, traitement de texte, tableur et graphisme seront disponibles en option.

Les grandes dates de Commodore



1982
Longtemps attendu, le **CBM 700** doit remplacer le 8032 et permettre à Commodore de se maintenir sur le marché des ordinateurs de bureau.



1981
Le **CBM 8032** offrait un affichage de 80 caractères, ce qui lui permettait de faire tourner des logiciels commerciaux sophistiqués.



1984
Le **SX-64** est une version portable du 64, dotée d'un écran couleur et d'un lecteur de disquette intégré.



1983
Le **Commodore 64** met fin aux restrictions dont souffrait le Vic, en proposant un affichage de 40 caractères et 64 K de mémoire.



1977
Le **PET** fut le premier micro-ordinateur grand public. Il se vend toujours très bien, après de nombreuses améliorations.



1979
Le **Vic-20**, premier micro-ordinateur Commodore très bon marché, a réussi sa percée en dépit de ses limites et d'une très forte concurrence.



1980
Le **SuperPET** (ou **CBM 9000**) était une tentative de modifier totalement le PET original afin d'en faire une machine de bureau.

PROGRAMME N° 17

QUELQUES PETITS PROGRAMMES SCOLAIRES EN MATHÉMATIQUES

Nous poursuivons la description de programmes relatifs à des problèmes mathématiques en essayant de créer cette fois un petit programme simple pour décomposer un nombre entier en facteurs premiers.

Exemples :

$$8 = 2^3$$
$$72 = 3^2 \times 2^3$$

Pour cela, on va utiliser une boucle pour tenter de diviser le nombre testé X par tous les entiers compris entre 0 et X/2.

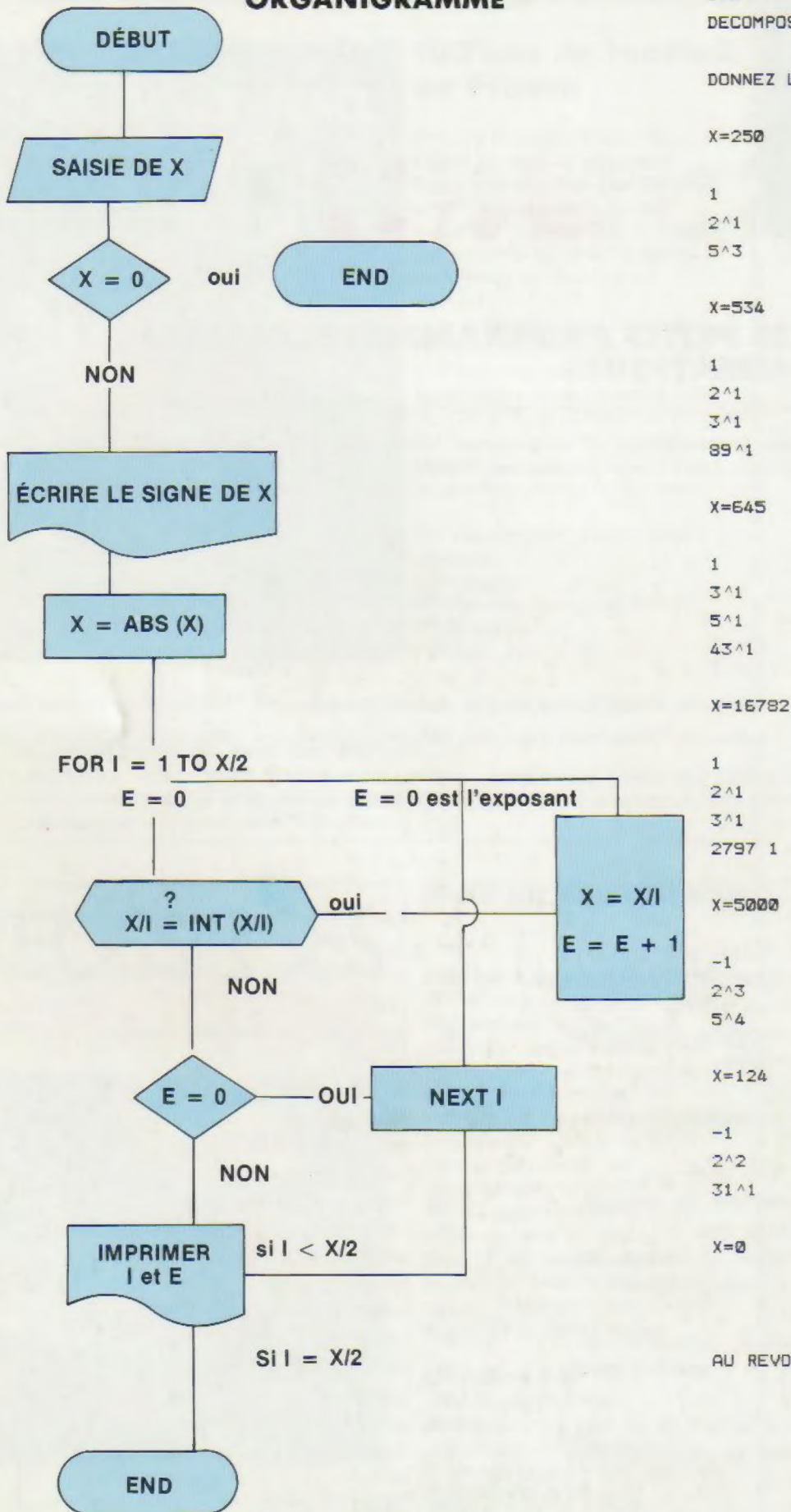
A chaque boucle on inscrira le facteur trouvé et sa puissance ou son exposant à l'aide de la notation « ^ » :

$$2^3 \text{ s'écrira } 2 \wedge 3$$
$$3^2 \times 2^3 = 3 \wedge 2$$
$$2 \wedge 3$$

Ici, nous pourrons tester des entiers relatifs en prenant en compte le signe comme un facteur. On utilisera, comme pour le programme précédent, la saisie de 0 pour stopper le programme.

```
5 HOME
10 INVERSE : PRINT "DECOMPOSITIO
   N EN FACTEURS PREMIERS": NORMAL
15 PRINT : PRINT : PRINT : PRINT
   "DONNEZ L'ENTIER A DECOMPOSE
   R (0 POUR STOP)"
20 PRINT : INPUT "X=":X
25 REM POUR STOPPER LE PROGRAMM
   E
30 IF X = 0 THEN GOTO 200
35 REM LE SIGNE EST UN FACTEUR
40 PRINT : PRINT SGN (X)
45 REM ON UTILISE LA VALEUR ABS
   OLUE DE X POUR LA SUITE
50 X = ABS (X)
55 REM BOUCLE DE RECHERCHE DES
   DIVISEURS DE X COMPRIS ENTRE
   2 ET X/2
60 REM AU DELA DE X/2 ON NE POU
   RRA OBTENIR DE NOUVEAUX DIVI
   SEURS DE X
65 FOR I = 2 TO X / 2
70 E = 0
75 IF X / I INT (X / I) THEN
   100
80 REM I EST UN FACTEUR PREMIER
85 X = X / I
90 E = E + 1
95 GOTO 75
99 REM IMPRIMER LE FACTEUR PREM
   IER TROUVE
100 IF E = 0 GOTO 150
105 REM IMPRIMER LES FACTEURS A
   VEC LEURS EXPOSANTS SOUS LA
   FORME A^B
110 PRINT I:"^":E
150 NEXT I
180 REM ON RECOMMENCE
185 GOTO 20
200 PRINT : PRINT : PRINT : PRINT
   "AU REVOIR FIN DE TRAITEMENT
   "
205 END
```

ORGANIGRAMME



1RUN

DECOMPOSITION EN FACTEURS PREMIERS

DONNEZ L'ENTIER A DECOMPOSER (0 POUR STOP)

X=250

1
2^1
5^3

X=534

1
2^1
3^1
89^1

X=645

1
3^1
5^1
43^1

X=16782

1
2^1
3^1
2797 1

X=5000

-1
2^3
5^4

X=124

-1
2^2
31^1

X=0

AU REVDIR FIN DE TRAITEMENT